Proceedings of the 26th Benelux Conference on Artificial Intelligence

BNAIC

Nijmegen, November 6-7, 2014

Franc Grootjen, Maria Otworowska, Johan Kwisthout (eds.)

BNAIC 2014

Benelux Conference on Artificial Intelligence

PROCEEDINGS OF THE TWENTY-SIXTH

BENELUX CONFERENCE ON ARTIFICIAL INTELLIGENCE

Nijmegen, November 6-7, 2014

Franc Grootjen, Maria Otworowska and Johan Kwisthout (eds.)

Franc Grootjen, Maria Otworowska and Johan Kwisthout

Benelux Conference on Artificial Intelligence (BNAIC) 2014
Proceedings of the 26^{th} Benelux Conference on Artificial Intelligence
Franc Grootjen, Maria Otworowska and Johan Kwisthout (eds.)
Nijmegen, 6-7 November 2014

ISSN 1568-7805

Cover: Nijmegen Bridge 'De Oversteek' (The Crossing), named after the heroic military action in World War II where American soldiers of the 504th Parachute Infantry Regiment attached to the 82nd Airborn Division crossed the river Waal on 20 September 1944.
Printing and binding: Ipskamp B.V.

Preface

This book contains the proceedings of the 26^{th} edition of the Benelux Conference on Artificial Intelligence. The conference was organized by the Radboud University. The multidisciplinarity of the field of Artificial Intelligence is visible by the number of research institutes and educational programs from Nijmegen that helped organizing this conference. Researchers from the Donders Institute for Brain, Cognition and Behaviour, the Institute for Computing and Information Sciences, the Centre for Language Studies and SSN Adaptive Intelligence together with lecturers from Artificial Intelligence, Computer Science and Communication and Information Sciences combined their efforts to organize this conference. As usual, the BNAIC was organized under the auspices of the Belgian-Dutch Association for Artificial Intelligence (BNVKI) and the Dutch Research School for Information and Knowledge Systems (SIKS).

The conference aims at presenting an overview of state-of-the-art research in Artificial Intelligence in Belgium, Luxemburg and the Netherlands, but does not exclude contributions from other countries. The received submissions show that AI researchers in the Benelux continue to work actively in many different areas of Artificial Intelligence and are open for new developments in technology and society.

To improve the exchange of ideas between AI researchers in the Benelux we preserved the tradition of accepting more than just regular papers describing original work: like previous years, we accepted extended abstracts describing both recent work published elsewhere as well as demonstrations of intelligent software. Moreover, to allow potential new researchers to get a taste for science, this year we added a special 'Thesis track', in which students could submit an extended abstract of their Bachelor or Master thesis. Students that were accepted in this track were invited to prepare a poster presentation of their work.

We received 67 submissions consisting of 24 regular full papers, 28 short papers, and 6 system demonstrations and 9 thesis abstracts. We are grateful to the Program Committee members who carefully reviewed all submissions. The program chairs made the final decisions. The overall acceptance rate was 87% (67% for full papers, 96% for compressed papers, 83% for demos and 100% for thesis abstracts).

There were many people involved in the organization of this conference. Apart from the local organization team we would like to thank all student volunteers, adminstrative and secretarial assistants and our sponsors. We gratefully acknowledge help from BNVKI board members and previous organizers. Finally, we thank our invited speakers, Simon Colton (Professor of Computational Creativity at Goldsmiths University of London) and Marc Pollefeys (Head of the Institute for Visual Computing at ETH Zürich).

October 2014

Franc Grootjen
Maria Otworowska
Johan Kwisthout

Organization

Program Chairs

Franc Grootjen
Johan Kwisthout
Antal van den Bosch
Perry Groot

Local Organization Team

Pashiera Barkhuysen	Franc Grootjen	Johan Kwisthout	Annet Wanders
Antal van den Bosch	Tom Heskes	Maria Otworowska	
Peter Desain	Lieve Jacques	Jolanda Rozenboom	
Perry Groot	Florian Kunneman	Suzan Verberne	

Program Committee

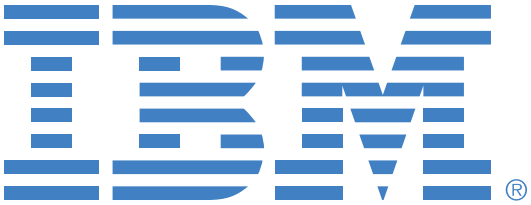
Huib Aldewereld	Francien Dechesne	Walter Kosters	Dirk Thierens
Reyhan Aydogan	Peter Desain	Johan Kwisthout	Leon van der Torre
Hendrik Blockeel	Frank Dignum	John-Jules Meyer	Karl Tuyls
Sander Bohte	Kurt Driessens	Peter Novàk	Jos Uiterwijk
Richard Booth	Jason Farquhar	Anne Nowé	Marina Velikova
Antal van den Bosch	Linda van der Gaag	Mykola Pechenizkiy	Joost Vennekens
Peter Bosman	Benjamin Gateau	Eric Postma	Katja Verbeeck
Tibor Bosse	Pascal Gribomont	Peter van der Putten	Sicco Verwer
Bruno Bouzy	Franc Grootjen	Jan Ramon	Arnoud Visser
Bert Bredeweg	Maaike Harbers	Nico Roos	Peter Vrancx
Egon van den Broek	Jaap van den Herik	Peter van Rosmalen	Louis Vuurpijl
Maurice Bruynooghe	Tom Heskes	Boris de Ruyter	Martijn Warnier
Martin Caminada	Koen Hindriks	Pierre-Yves Schobbens	Mathijs de Weerdt
Patrick De Causmaecker	Tom Holvoet	Johannes Scholtes	Gerhard Weiss
Tristan Cazenave	Arjen Hommersom	Evgueni Smirnov	Marco Wiering
Martine De Cock	Mark Hoogendoorn	Matthijs Spaan	Floris Wiesman
Tom Croonenborghs	Veronique Hoste	Ida Sprinkhuizen-Kuyper	Jef Wijsen
Walter Daelemans	Maurits Kaptein	Thomas Stuetzle	Mark Winands
Gregoire Danoy	Uzay Kaymak	Johan Suykens	Cees Witteveen
Mehdi Dastani	Tomas Klos	Anette ten Teije	Yingqian Zhang

Additional Reviewers

Sofie De Clercq	James Marquardt	Erwin Walraven
Charlotte Gerritsen	Lode Vuegen	Jeroen de Man

Sponsors

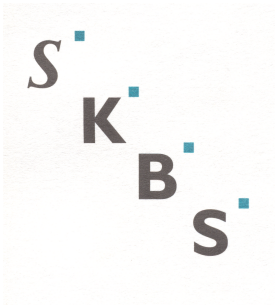

BNVKI
AIABN




NWO
Netherlands Organisation for Scientific Research


SNN
ADAPTIVE
INTELLIGENCE


Donders Institute
for Brain, Cognition and Behaviour


S
K
B
S


textkernel


Radboud University
IN DEI NOMINE FELICITER

Contents

Full papers

Learning to Recognize Horn and Whistle Sounds for Humanoid Robots 1
Niels Backer and Arnoud Visser

Combining Combinatorial Game Theory with an α-β Solver for Domineering 9
Michael Barton and Jos Uiterwijk

Using Distances for Aggregation in Abstract Argumentation 17
Richard Booth and Mikołaj Podlaszewski

Grounded Semantics and Infinitary Argumentation Frameworks 25
Martin Caminada and Nir Oren

Capturing Evidence and Rationales with Requirements Engineering and Argumentation-Based Techniques 33
Sepideh Ghanavati and Marc van Zee

Fast Laplace Approximation for Gaussian Processes with a Tensor Product Kernel 41
Perry Groot, Markus Peters, Tom Heskes and Ketter Wolf

Predicting Pseudo-Random Behaviour in Professional Sports 49
Manuel Kauschinger and Kurt Driessens

Gesture Detection and Recognition by Repetition for Expressive Control 57
Bas Kooiker and Makiko Sadakata

Event detection in Twitter: A machine-learning approach based on term pivoting 65
Florian Kunneman and Antal van Den Bosch

Case-driven Agent-based Simulation: a Methodology for the Interdisciplinary Development and Evaluation of Cognitive Architectures 73
Samer Schaat and Dietmar Dietrich

Stratied Action Negation, a Logic about Travel 81
Xin Sun and Huimin Dong

How do Pessimistic Agents save Miners? A STIT Based Approach 88
Xin Sun, Zohreh Baniasadi and Shuwen Zhou

Dynamic Lateral Stability for an Energy Efficient Gait 95
Zhenglong Sun and Nico Roos

Valuation of Cooperation and Defection in Small World Networks: A Behavioral Robotic Approach 103

Bijan Ranjbar-Sahraei, Irme M. Groothuis, Karl Tuyls and Gerhard Weiss

Validating Ontologies for Question Generating 111
Marten Teitsma, Jacobijn Sandberg, Bob Wielinga and Guus Schreiber

Monte-Carlo Tree Search for Poly-Y 119
Lesley Wevers and Steven Te Brinke

Compressed papers

Fair-Share ILS: A Simple State of the Art Iterated Local Search Hyperheuristic 129
Steven Adriaensen, Tim Brys and Ann Nowé

On the Input/Output Behavior of Argumentation Frameworks 131
Pietro Baroni, Guido Boella, Federico Cerutti, Massimiliano Giacomin, Leon van der Torre and Serena Villata

Simplifying the Visualization of Confusion Matrix 133
Emma Beauxis-Aussalet and Lynda Hardman

LOD Laundromat: A Uniform Way of Publishing Other People's Dirty Data 135
Wouter Beek, Laurens Rietveld, Hamid Bazoobandi, Jan Wielemaker and Stefan Schlobach

Towards Aggression De-escalation Training with Virtual Agents: A Computational Model 137
Tibor Bosse and Simon Provoost

Combining Multiple Correlated Reward and Shaping Signals by Measuring Confidence 139
Tim Brys, Ann Nowé, Daniel Kudenko and Matthew E. Taylor

Decomposition of Intervals in the Space of Anti-Monotonic Functions 141
Patrick De Causmaecker and Stefan De Wannemacker

Probabilistic Argumentation Frameworks - A Logical Approach 143
Dragan Doder and Stefan Woltran

Structural Properties as Proxy for Semantic Relevance in RDF Graph Sampling 145
Rinke Hoekstra, Laurens Rietveld, Stefan Schlobach and Christophe Guéret

Feasibility Estimation for Clinical Trials 147
Zhisheng Huang, Frank Van Harmelen, Annette Ten Teije and Andre Dekker

Virtual Reflexes 149
Catholijn Jonker, Joost Broekens and Aske Plaat

Autonomous E-Coaching in the Wild: Empirical Validation of a Model-Based Reasoning System . 151
Bart Kamphorst, Michel Klein and Arlette van Wissen

Market Garden: a Simulation Environment for Research and User Experience in Smart Grids 153
Bart Liefers, Felix N. Claessen, Eric Pauwels, Peter A.N. Bosman and Han La Poutré

Measuring Diversity of Preferences in a Group 155
Vahid Hashemi and Ulle Endriss

Multi-objective Gene-pool Optimal Mixing Evolutionary Algorithms 157
Ngoc Hoang Luong, Han La Poutré and Peter Bosman

A Successful Broker Agent for Power TAC 159
Bart Liefers, Jasper Hoogland and Han La Poutré

Smarter smartphones: understanding and predicting user habits from GPS sensor data 161
Giuseppe Maggiore, Carlos Santos and Aske Plaat

Efficient Heuristics for Power Constrained Planning of Thermostatically Controlled Loads 163
Frits de Nijs, Mathijs de Weerdt and Matthijs Spaan

Nash Equilibria in Shared Effort Games 165
Gleb Polevoy, Stojan Trajanovski and Mathijs de Weerdt

A Novel Population-based Multi-Objective CMA-ES and the Impact of Different Constraint Handling Techniques 167
Sílvio Rodrigues, Pavol Bauer and Peter Bosman

Bounded Approximations for Linear Multi-Objective Planning under Uncertainty 169
Diederik Roijers, Joris Scharpff, Matthijs Spaan, Frans Oliehoek, Mathijs De Weerdt and Shimon Whiteson

Combining Simulated Annealing and Monte Carlo Tree Search for Expression Simplification 171
Ben Ruijl, Jos Vermaseren, Aske Plaat and H.Jaap Van den Herik

Combining Model-Based EAs for Mixed-Integer Problems 173
Krzysztof Sadowski, Dirk Thierens and Peter Bosman

Anchor-Profiles: Exploiting Profiles of Anchor Similarities for Ontology Mapping 175
Frederik Schadd and Nico Roos

Causal discovery from databases with discrete and continuous variables 177
Elena Sokolova, Perry Groot, Tom Claassen and Tom Heskes

Flexibility and Decoupling in Simple Temporal Networks 179
Michel Wilson, Tomas Klos, Cees Witteveen and Bob Huisman

Robot Mood is Contagious: Effects of Robot Body Language in the Imitation Game 181
Junchao Xu, Joost Broekens, Koen Hindriks and Mark Neerincx

Demos

Using Facial Expressions for Personalised Gaming 185
Paris Mavromoustakos Blom, Sander Bakkes and Diederik Roijers

Teaching Mario. Demonstrating the Effectiveness of Human Guidance when q-learning 187
Roland Meertens

Enhancing operational work in maritime safety-and-security tasks 189
Steffen Michels, Marina Velikova, Bas Huijbrechts, Peter Novak, Jesper Hoeksma, Roeland Scheepens, Jan Laarhuis and Andre Bonhof

An Implementation for Distances between Labellings in Abstract Argumentation 191
Mikołaj Podlaszewski and Yining Wu

Interpreting EEG Signals using Artificial Intelligence 193
Felipe Gomez Marulanda, Ann Nowe, Yann-Michael De Hauwere and Peter Vrancx

Thesis abstracts

The Detection Of Facial Expressions for Action Coordination 199
Tessa Beinema, Ron Dotsch and Franc Grootjen

Modeling and forecasting elections using topic models 201
Bas van Berkel

Using Neighbourhood-based Collaborative Filtering to Predict E-Learning Exercise Difficulty 203
Floris Devriendt, Ruben Lagatie, Maarten Devillé, Peter Vrancx and Ann Nowé

Studying Social Interactions using Swarm Robotics 205
Irme M. Groothuis and Bijan Ranjbar-Sahraei

Adaptive Learning Using the Exclusion Principle 207
Iris Monster, James M. McQueen and Peter Desain

Rotation invariant feature extraction in the classification of galaxy morphologies 209
Steven Reitsma

Broad-Band Visually Evoked Potentials: Towards Enhanced Speller BCIs 211
Jordy Thielen, Philip van den Broek, Jason Farquhar and Peter Desain

Finding Substitutions of Rare Earth Elements Using Publication Data 213
Kirill I. Tumanov

Traffic Flow Optimization using Reinforcement Learning 215
Erwin Walraven

Part I

Full papers (A)

Learning to recognize horn and whistle sounds for humanoid robots

Niels W. Backer Arnoud Visser

University of Amsterdam, P.O. Box 94323, 1090 GH Amsterdam

Abstract

The efficiency and accuracy of several state-of-the-art algorithms for real-time sound classification on a NAO robot are evaluated, to determine how accurate they are at distinguishing horn and whistle sounds in both optimal conditions, and a noisy environment. Each approach uses a distinct combination of an audio analysis method and a machine learning algorithm, to recognize audio signals captured by NAO's four microphones. A short summary of three audio analysis preprocessing methods is provided, as well as a description four machine learning techniques (Logistic Regression, Stochastic Gradient Descent, Support Vector Machine, and AdaBoost-SAMME) which could be used to train classifiers which would distinguish whistle and horn signals from background noise. Experimental results show that for each of the acquired data sets, there are multiple high-accuracy solutions available. Actually, the accuracy and precision results were all so high, that a more challenging dataset is needed to determination which method is optimal for this application.

1 Introduction

Although a large portion of AI robotics research is focused mostly on topics of high-level cognitive tasks, like path planning and robot mapping & localization, one of the most important aspects of natural intelligence is its ability to react to trigger-events in the environment. Fluent interaction with environmental signals is not only a requirement for effective Human-Robot Interaction (HRI), but also an important component for systems that have real-world applications. For these reasons, a sound recognition challenge was issued this year by the RoboCup Standard Platform League (SPL) technical committee, which organizes the yearly robot football world cup. The goal of this challenge is to make the humanoid NAO robots used in the SPL recognize audio signals, more specifically predefined start/stop signals and whistle signals, emitted from the sideline of the pitch.

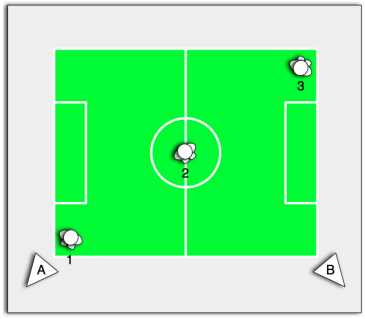


Figure 1: The NAO audio recognition challenge setup. Predefined signals will be emitted in alternating sequences from speakers A and B.

Since the RoboCup is a very noisy event, and NAO's microphones are quite small and of mediocre quality, this challenge is not very straightforward. Because audio classification has been plagued by noise in the past, this challenge calls for a qualitative assessment of a range of techniques applied to signal recognition.

Most digital audio signal processing techniques have a similar method structure. Firstly, the audio signal is converted from analog to digital, after which signal processing techniques are applied to convert the signal to a more compact representation, which contains the features which could be recognized. A classic example of such filter is the Fast Fourier Transform (FFT), which is a very computationally efficient method to find the spectral properties of the signal. An alternative method is the Discrete Wavelet Transform (DWT), which extracts a wavelet power spectrum. Similar to the FFT, DWT can be used to extract frequencies from a signal, but has a distinct advantage for analyzing temporal sequences: It captures both frequency and location (in time) information. The last audio-analysis technique covered in this study is Mel Frequency Cepstral Coefficients (MFCC). Audio analysis using the Cepstral representation of the short-term power spectrum can be used to identify pitch and formant frequencies in speech, and has in fact become a very conventional way to process human speech before recognition.

The second phase is the classification (or recognition) step. While a large number of applications for classifying simple stationary signals have used a frequency mask or single-layer perceptron, the aim of this study is to provide a performance comparison of machine learning algorithms as well. The optimal outcome would be to find a configuration which is impervious to noise, which is a considerable obstacle for common classification methods. In order to combat the effect of noise on audio recognition, we will inspect classification algorithms, some well-established, some quite new, such as logistic regression, stochastic gradient descent, support vector machines, and AdaBoost-SAMME.

The study is organized as follows: Section 2 describes related work. Section 3.1 describes how five datasets were recorded with the Nao robots. Section 3.2 thereafter a short summary of audio analysis techniques is given. Section 3.3 will discuss the machine learning techniques, after which research methods will be discussed, followed by results, a discussion, and notes on future work.

2 Related Research

Sound recognition in robots is one of the major steps in creating autonomous systems that can directly interact with humankind. According to the RoboCup Standard Platform League challenges set out for 2014[1], there is a number of reasons to eliminate WLAN communication within RoboCup soccer matches, with 'not very human-like' being the most important for the Artificial Intelligence field.

There have been previous attempts at tackling the recognition of whistles and predefined horn signals [2], also considering noisy environments [11]. These systems have been proven to be reliable, though they only consider one possible system architecture, using common methods. Both solutions utilize the Fast Fourier Transform (FFT), a frequency mask, and a single-layer perceptron. However, a perceptron without hidden layers can only be trained to recognize a small set of patterns, since its architecture only allows for it to discern instances in models with linearly separable patterns. Attempts at phoneme/speech recognition have successfully used neural networks as an alternative to simple perceptrons [8].

Some speech and dialogue based systems for the NAO also exist. An example of such system is the event-based conversational system for Human-Robot Interaction (HRI) [10]. Though not yet fully autonomously usable, it provides a solid basis for future research. One of the biggest issues in robot audio recognition is background noise. HRI approaches using the NAO built-in conversational system have had some difficulties getting around this barrier [9]. However, more recent research with a fairly simple approach, using the FFT of a signal and logistic regression, has obtained very promising results in noisy environments [14].

3 Research Method

In this section, a description will be given of all methods, their implementation details, and optimizations for the task in hand. Firstly, we will discuss data collection on the NAO robot, in particular the several data sets that were recorded for this study. In the next section all the audio-analysis methods will be set out, as well as the instance vectors they produce for the machine learning phase. A more extensive description of the audio-analysis techniques is available in [1]. This paper will focus on the machine learning methods available to train the classifiers.

[1] http://www.informatik.uni-bremen.de/spl/pub/Website/Downloads/Challenges2014.pdf

3.1 Data Collection

All experiments were conducted using audio recorded indoors, on the NAO robot. The types of audio consist of two predefined horn signals, namely a 'start' and a 'stop' signal, both a sequence of $200Hz$ and $320Hz$ sinusoids, and a complex whistle sound, as used by a referee in any regular football match, which has a broad peak between 6 and $7kHz$. The distance between the robot and the audio source, as well as the level of background noise, were both decided upon after preliminary audio data assessment: the signal should be audible to a human being.

Five data sets were recorded using NAO's microphone array. The SPL challenge requires the recognition of both predefined horn signals and a whistle, brought by the team itself. For both types of signals, two distinct datasets were recorded. Each dataset consists of two recordings, one with a low amount of noise and high signal strength (sets 2 and 4), the other with background noise and a low signal strength, recorded further away from the signal source (sets 1 and 3). Some background noise, containing small amounts of human speech, was also recorded, to serve as the negative samples for training (set 10).

Table 1: Data set description. All data sets were recorded at a 48000 Hz sample rate, using pcm16 encoding, on all 4 NAO microphones simultaneously.

Set	Description	# audio samples	Duration (s)
1	Whistle set, with background noise	4450	56
2	Whistle set, minimal noise	4205	53
3	Horn signal set, with background noise	2210	28
4	Horn signal set, minimal noise	2225	28
5	Whistle data, sets 1 and 2 combined	3289	41
6	Horn signal data, sets 3 and 4 combined	4436	55
7	Noisy data, sets 1 and 3 combined	6661	83
8	Clean data, sets 2 and 4 combined	6430	80
9	All other sets combined	7726	97

The microphones on the NAO provide a 48 kHz audio stream from 4 microphones, situated at the front, rear, and sides of the head. Audio frames were taken at a frequency of 20 Hz, or every 50 ms. The recordings had duration between 28 and 97 seconds. The recordings should not contain any clipping, so periods of silence between actual signals were scrubbed, resulting into between 2210 and 7726 samples where an audio signal was present. All those audio samples were manually marked as positive, $y = 1$, or negative, $y = 0$, to distinguish the target signal from the background noise. Combined data sets were then created, namely sets 5, 6, 7, and 8, one for each type of signal, and one per amount of background noise, by concatenating data from the first four sets into a single file. Finally, a data set containing all types of audio was created, namely set 9.

3.2 Audio Analysis Methods

The datesets contained at least 2000 audio samples, which is a sufficient size not to skip any samples because of the analysis processing time, and more than enough samples to produce a precise representation of the signal. These audio samples were normalized to reduce the effect of the audio volume on the final representation.

3.2.1 Fast Fourier Transform

There is a plethora of FFT algorithms, each of which calculates the Discrete Fourier Transform, or DFT, of a signal. The DFT is a discretized form of a method for expressing a function as a sum of periodic components, and for recovering the function from these components. Using an input signal of N samples, the DFT is defined as

$$X_k = \sum_{n=0}^{N-1} x_n e^{-2\pi i \frac{kn}{N}} \qquad k = 0, \ldots, N-1$$

For this study, the Scientific Python (SciPy) implementation of FFT was used[2], specifically `scipy.fftpack.rfft`. This Real Fast Fourier Transform is an implementation of an adapted form of the Cooley– Tukey algorithm [5], which can only be applied to a series of real values, such as a sound signal. By ignoring the complex conjugates, which would yield negative frequencies, it gains a considerable speed boost. This method yields $\frac{N}{2} + 1$ coefficients, of which the log values are taken, which are more suitable for most machine learning algorithms. The FFT is truncated at around 12 kHz, since early analysis shows that higher frequencies are redundant for the task of whistle recognition. This process generates the 680-feature length vectors used for training and testing our classifiers.

3.2.2 Discrete Wavelet Transform

For this study, the implementation of Discrete Wavelet Transform from the open source PyWavelets Python module, `pywt.WaveletPacket`, was used[3]. Like most implementations of DWT, it calculates the transform of a discrete signal by passing it through a series of filters, using convolution. Instead of analyzing sinusoid levels in a signal, DWT attempts to find wavelets, which are wave-like oscillations with an amplitude that begins at zero, increases, and then decreases back to zero. There are many wavelet families, the particular wavelet used here is commonly referred to as the Daubechies 4-tap wavelet, or db4 [6]. Firstly, the signal is decomposed by simultaneously applying a low pass filter and a high pass filter, which output the approximation and detail coefficients, respectively:

$$y_{low}[n] = \sum_{k=-\infty}^{\infty} x[k]h[2n-k]$$

$$y_{high}[n] = \sum_{k=-\infty}^{\infty} x[k]g[2n-k]$$

Where h and g are the low and high filter's impulse responses, respectively. The output of the filters is then subsampled, because according to Nyquist's rule half of the samples can be discarded since half of the frequency band is removed. Hereafter the process of decomposition and subsampling is repeated in a filter bank. The repetition depth, or level, determines the frequency resolution; higher-level coefficients have a high frequency resolution, and a low time resolution. Preliminary analysis shows that a level 4 DWT provides a suitable representation for classification, as well as the fact that the higher-level coefficients contain no useful information. Therefore, the DWT is truncated at the fifth coefficient level, out of 16 levels in total. For each 2400 audio samples, the mean of the resulting 44 coefficient vectors is taken, to produce a 5-feature length vector for classification. Though this may seem to be a small amount of features, results show that it is sufficient for a classifier to attain a very high accuracy.

3.2.3 Mel Frequency Cepstral Coefficients

The Mel-Frequency Cepstral is a representation of the short-term power spectrum of a sound, based on a linear cosine transform of a log power spectrum on a nonlinear Mel scale of frequency. MFCCs are commonly used as features in speech and speaker recognition systems, since its warped frequency scale approximates the human auditory system's response more closely than the linearly-spaced frequency bands of the FFT or common cepstral representation. For this study, the following approach was used to produce MFCCs:

1. Compute the FFT for an audio sample.
2. Map the obtained FFT onto the Mel scale, using triangular overlapping mathematical windows, also known as producing a triangular bank filter.
3. Compute the log spectrum. This is done by taking the logs of the powers at each of the Mel frequencies.
4. Finally, compute the discrete cosine transform of the list of Mel log powers. This produces the MFCCs.

[2] `http://www.scipy.org/`
[3] `http://www.pybytes.com/pywavelets/`

This method of calculating coefficients allows you to pick the size of the MFCC array as you choose. Result analysis shows that all characteristic components of both the horn and whistle signals can be captured in a 13-feature length instance vectors, which was the design choice for this study.

3.3 Machine Learning Algorithms

The machine learning algorithms used in this study were implemented in the Scikit-Learn Python module [13]. Before classification, all analyzed data was scaled in order to produce data with zero mean and unit variance, which is needed for algorithms like support vector machines. This scaling was first applied to the training set, and its parameters were then used to scale the test set, in the same manner.

The data sets were then shuffled, and split with 70% of the data for the training set, and 30% reserved for cross-validation. These examples can be represented as a set of instance vectors $\vec{x}$, and mark y: $\{(\vec{x}_i, y_i)\}_{i=1}^{M}$, where $\vec{x}_i \in \mathbb{R}^M$ and $y_i \in \{0, 1\}$. The amount of samples, M, greatly exceeds the amount of features N: $M \gg N$.

The results in this paper were obtained after each classification algorithm was run 20 times independently. For each classifier run, the data was re-shuffled and split, after which the classifiers were fitted. Their accuracy, precision, recall, F1-score, false positives, false negatives, true positives, and true negatives were stored and averaged over all of the runs.

3.3.1 Logistic Regression

The most efficient algorithm in terms of computing power used for this study is the logistic regression binary classifier with l^2 normalization [3]. This particular form of regularization was selected since previous approaches to this task yielded promising results [14]. Its hypothesis space consists of sigmoid functions, $h_\theta(\vec{x}) = \frac{1}{1+e^{-\vec{\theta}^T\vec{x}}}$, where $\vec{\theta}$ are the adjustable weights, and $\vec{x}$ the sample vectors. In its implementation, $y_i \in \{-1, 1\}$, yet this is solved by automatically setting all instances initially marked 0 to -1 prior to fitting, and resetting them for the evaluation phase. It solves the following unconstrained optimization problem:

$$min_{\vec{\theta}} \left(\frac{1}{2}\vec{\theta}^T\vec{\theta} + C\sum_{i=1}^{N} \xi(\vec{\theta}; \vec{x}_i, y_i) \right)$$

Where ξ is the loss function:

$$\xi(\vec{\theta}; \vec{x}_i, y_i) = log(1 + e^{-y_i\vec{\theta}^T\vec{x}_i})$$

In order to combat overfitting, but retain a high accuracy rate, a C-sweep (inverse regularization, $C = \frac{N}{\alpha}$) was performed.

3.3.2 Stochastic Gradient Descent

The SGD classifier used here is a first-order learning routine, which, in contrast to batch gradient descent, considers a single instance vector at a time. The change of its weights is given by the gradient of the classification error, which is evaluated only for the latest in the training sample sequence. This variant is sometimes known as the On-line Gradient Descent algorithm (OGD). Its weights vector $\vec{\theta}$ is updated like so:

$$\vec{\theta} \leftarrow \vec{\theta} - \eta\nabla Q_i(\vec{\theta})$$

Where $Q_i(\vec{\theta})$ is the value of the loss function at example i, and the adaptive learning rate η is given by

$$\eta^{(t)} = \frac{1}{\alpha(t_0 + t)}$$

Where t is the weight number, $t = 1, 2, \ldots, T$. For the task of audio recognition, the decision was made to compute the gradient over a simple hinge loss function. This linear approach has previously successfully been applied to audio recognition, for instance in music annotation [12].

3.3.3 AdaBoost-Samme

AdaBoost-SAMME [15] is a state-of-the-art generalization of the AdaBoost algorithm [7]. This meta-estimator begins by fitting a classifier, a decision tree classifier in this case, on the original dataset, and then fits additional copies of the classifier on the same dataset, but where the weights of incorrectly classified instances are modified such that subsequent classifiers focus more on difficult cases.

The 'weak classifier' used for this algorithm, a decision tree classifier, is first built using a greedy, top-down recursive partitioning strategy. It uses the Gini impurity function to calculate the quality of a split, which is a measure of how often a randomly picked instance would be incorrectly labeled if it were randomly labeled according to the distribution of labels it has already encountered. On its own, this decision tree classifier achieves a very low accuracy, but using AdaBoost with a standard learning rate of 1.0, the accuracy and F1-scores soar.

3.3.4 Support Vector Machine/C-Support Vector Classifier

The support vector machine used in this study is the C-support vector classifier `sklearn.svm.SVC`; an l^2-regularized SVM. In its implementation, $y_i \in \{-1, 1\}$, yet this is solved by automatically setting all instances initially marked 0 to -1 prior to fitting, and resetting them for the evaluation phase. Similar to logistic regression, this support vector machine solves the following problem:

$$min_{\vec{\theta}} \left(\frac{1}{2}\vec{\theta}^T\vec{\theta} + C\sum_{i=1}^{N} \xi(\vec{\theta}; \vec{x}_i, y_i) \right)$$

Yet with a loss function different from logistic regression, ξ, defined as

$$\xi(\vec{\theta}; \vec{x}_i, y_i) = max(1 - y_i\vec{\theta}^T\vec{x}_i, 0)^2$$

For FFT and MFCC, the error term penalty parameter $C = \frac{N}{\alpha}$ was kept at 1.0 after preliminary assessment - lowering C made model convergence unfeasible, while higher values gravely affected F1-scores. For DWT analysis, C was adjusted to 1.4. In the decision function used for FFT and MFCC, SVC uses a sign function with a third degree polynomial kernel K, without an intercept term:

$$h_\theta(\vec{x}) = sgn\left(\sum_{i=1}^{N} y_i\alpha_i K(\vec{x}_i, \vec{x}) \right)$$

Note that for most types of audio analysis, support vector machines will yield very poor results if the data has not been scaled. SVC is an inefficient algorithm for large data sets, since its fit time complexity is more than quadratic. In order to speed up convergence, a shrinking heuristic was used. This shrinking heuristic removes certain elements that may have already been bounded during the decomposition iterations [4]. Because of the optimizations to this algorithm, convergence time was brought down to a feasible level, while retaining a low criterion for stopping tolerance ($tol = 0.001$).

4 Results

All analysis/classifier combinations were run 20 separate times on each combined data set, with a cross validation stage on the remaining 30% of the data that was not used during the training stage. The accuracy score reflects that of a single classifier, run on all audio samples in from all 4 microphone inputs, unless specified otherwise.

The first of experiments was performed on both signal types separately, in order to ascertain any discrepancies in performance. The results can be found in Table 2.

These results show that although classifier performance varies slightly between different types of audio, it is not clear whether this is due to the signal type itself, or to small imperfections in data set 6, since there is only a 0.2% accuracy rate variation.

In order to obtain a clear view of the difference in classifier performance in varying conditions, each classifier/analysis combination was run on both the noisy and clear data sets. These results show something unexpected: None of the analysis/classifier combinations seems to be heavily affected by signal noise. The accuracy rates on combined sets 7 and 8 (see table 3), show only minor variations.

	FFT	DWT	MFCC
SVM	99.97	**100.00**	**100.00**
SGD	99.99	99.99	**100.00**
ADA	99.97	**100.00**	99.97
LogReg	**100.00**	99.98	99.99

	FFT	DWT	MFCC
SVM	99.73	99.80	99.87
SGD	99.70	99.82	**99.90**
ADA	99.87	99.84	99.87
LogReg	99.88	99.83	99.87

Table 2: Accuracy percentage scores on the whistle data set (set 5 - left) and horn data set (set 6 - right).

	FFT	DWT	MFCC
SVM	97.44	94.99	99.60
SGD	98.74	96.25	99.22
ADA	99.69	98.99	99.65
LogReg	**99.78**	96.35	99.21

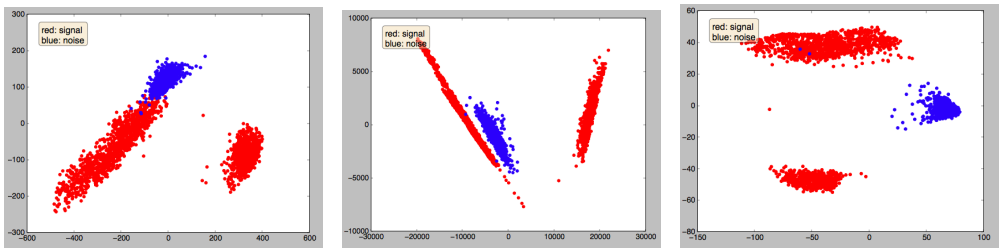
	FFT	DWT	MFCC
SVM	98.66	96.28	99.90
SGD	99.49	96.92	97.88
ADA	**99.88**	**99.88**	99.87
LogReg	99.87	97.88	99.87

Table 3: Accuracy percentage scores on the combined noisy (set 7 - left) and clean data (set 8 - right).

Table 3 shows a performance comparison between the total combined noisy and clean sets. Note that these sets contain multiple types of signals - start, stop, and whistle signals. Their overall performance decreases slightly when applied to noisy data, but only by a small margin. This implies all methods are mostly impervious to background noise.

4.1 Evaluation

The purpose of this experimental study is to assess several potential system proposals for the recognition of trigger-events in the form of audio signals, and to compare their performance in discerning both low-frequency sinusoids and a high-frequency whistle sound. Figures 2a, 2b, and 2c are scatter plots of $\sim 5\%$ of the processed versions of set 9, of which the dimensionality was reduced using Principal Component Analysis (PCA). These PCA representations clearly show that all audio analysis techniques provide a robust method for discerning each data point, since most of the sets are linearly separable.



(a) FFT (b) DWT (c) MFCC

Figure 2: PCA visualization of the combined data set 9 for several audio analysis techniques.

5 Conclusion

Twelve different combinations of audio analysis and machine learning techniques were studied for their effectiveness in recognizing three auditory triggers on a NAO robot using four microphones. The audio analysis was performed using Fast Fourier Transform, Discrete Wavelet Transform, and Mel Frequency Cepstral Coefficients, whereas the recognition task was handled by a C-Support Vector Classifier, Stochastic Gradient Descent, AdaBoost-SAMME, and Logistic Regression.

The results reported herein clearly prove that all twelve combinations are good candidates to solve the challenge of recognizing both the predefined horn signal with a sinusoid shape and an arbitrary whistle signal. Experiments were carried out comparing the accuracy rates, in order to discover the

optimal settings for each algorithm in both noisy and quiet environments. Performance was not severely affected by noise, and because these results are based on $50ms$ time frames, while a real signal would be at least several frames long, all of these methods possess the ability to recognize natural audio signals reliably.

References

[1] Niels W. Backer. Horn and whistle recognition techniques for nao robots. Bachelor thesis, Universiteit van Amsterdam, June 2014.

[2] Andrea Bonarini, Daniele Lavatelli, and Matteo Matteucci. A composite system for real-time robust whistle recognition. In *RoboCup 2005: Robot Soccer World Cup IX*, volume 4020 of *Lecture Notes in Computer Science*, pages 130–141. Springer Berlin Heidelberg, 2006.

[3] Bernhard E Boser, Isabelle M Guyon, and Vladimir N Vapnik. A training algorithm for optimal margin classifiers. In *Proceedings of the fifth annual workshop on Computational learning theory*, pages 144–152. ACM, 1992.

[4] Chih-Chung Chang and Chih-Jen Lin. Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27:1–27:27, 2011.

[5] James W. Cooley and John W. Tukey. An algorithm for the machine calculation of complex fourier series. *Mathematics of Computation*, 19(90):297–301, 1965.

[6] Ingrid Daubechies. *Ten lectures on wavelets*, volume 61 of *CBMS-NSF Series in Applied Mathematics*. SIAM Publications, Philadelphia, 1992.

[7] Yoav Freund and Robert E Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. In *Computational learning theory*, pages 23–37. Springer, 1995.

[8] Alex Graves, Abdel-Rahman Mohamed, and Geoffrey Hinton. Speech recognition with deep recurrent neural networks. In *IEEE International Conference onAcoustics, Speech and Signal Processing (ICASSP'13)*, pages 6645–6649, May 2013.

[9] Dinesh Babu Jayagopi, Samira Sheiki, David Klotz, et al. The vernissage corpus: A conversational human-robot-interaction dataset. In *8th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 149–150. IEEE Press, 2013.

[10] Ivana Kruijff-Korbayová, Georgios Athanasopoulos, Aryel Beck, et al. An event-based conversational system for the nao robot. In *Proceedings of the Paralinguistic Information and its Integration in Spoken Dialogue Systems Workshop*, pages 125–132. Springer New York, 2011.

[11] Gil Lopes, Fernando Ribeiro, and Paulo Carvalho. Whistle sound recognition in a noisy environment. In *Proceedings of Controlo'2010 - 9th Portuguese Conference on Automatic Control*, pages 172–179, September 2010.

[12] Juhan Nam, Jorge Herrera, Malcolm Slaney, and Julius O Smith. Learning sparse feature representations for music annotation and retrieval. In *Proceedings of 13th International Society for Music Information Retrieval Conference (ISMIR)*, pages 565–570, 2012.

[13] F. Pedregosa, G. Varoquaux, A. Gramfort, et al. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[14] Kyle Poore, Saminda Abeyruwan, Andreas Seekircher, and Ubbo Visser. Single- and multi-channel whistle recognition with nao robots. In *RoboCup 2014: Robot Soccer World Cup XVIII*, Lecture Notes on Artificial Intelligence series. Springer, Heidelberg, 2015. to be publicated.

[15] Ji Zhu, Hui Zou, Saharon Rosset, and Trevor Hastie. Multi-class adaboost. *Statistics and Its Interface*, 2(3):349—360, 2009.

Combining Combinatorial Game Theory with an α-β Solver for Domineering

Michael Barton Jos Uiterwijk

Department of Knowledge Engineering, Maastricht University
P.O. Box 616, 6200 MD Maastricht, The Netherlands

Abstract

Combinatorial games are a special category of games sharing the property that the winner is by definition the last player able to move. To solve such games two main methods are being applied. The first is a general NegaScout search with many possible enhancements. This technique is applicable to every game, mainly limited to the size of the game due to the exponential explosion of the solution tree. The second way is to use techniques from Combinatorial Game Theory (CGT), with very precise CGT values for (subgames of) combinatorial games. This method is only applicable to relatively small (sub)games. In this paper we show that the methods can be combined in a fruitful way by using endgame databases filled with CGT values.

We apply this technique to the game of Domineering, a well-known partisan type of combinatorial game. Our test suite consisted of all 36 non-trivial boards with dimensions from 2 to 7. Endgame databases were created for all subgames of size 15 and less. The CGT values were calculated using the CGSUITE package. We show how CGT values of subgames can be used in several ways as refinements of a basic NegaScout solver. Experiments reveal up to 99% reduction in number of nodes investigated.

1 Introduction

Domineering is a two-player perfect-information game invented by Göran Andersson around 1973. It was popularized to the general public in an article by Martin Gardner [7]. It can be played on any subset of a square lattice, though mostly it is restricted to rectangular $m \times n$ boards, where m denotes the number of rows and n the number of columns. The version introduced by Andersson and Gardner was the 8×8 board.

Play consists of the two players alternately placing a 1×2 tile (domino) on the board, where the first player may place the tile only in a vertical alignment, the second player only horizontally. Dominoes may not overlap. The first player being unable to move loses the game, his opponent (who made the last move) being declared the winner. Since the board is gradually filled, i.e., Domineering is a converging game, the game always ends, and ties are impossible. With these rules the game belongs to the category of *combinatorial games*, for which a whole theory (the Combinatorial Game Theory, CGT in short) has been developed, especially by Conway [6] and Berlekamp et al. in their famous book set *Winning Ways* [3]. In combinatorial game theory the first player conventionally is called Left, the second Right, though in our case we will use the more convenient indications of Vertical and Horizontal, for the first and second player, respectively.

Among combinatorial game theorists Domineering received quite some attention, but this was limited to rather small or irregular boards [1, 2, 3, 6, 9, 22]. Larger (rectangular) boards were solved using α-β search [10], leading to solving all boards up to and including the standard 8×8 board [4], later extended to the 9×9 board [21, 8], and finally extended to larger boards up to 10×10 [5].

In this article we describe the results of building endgame databases with exact CGT values for the game of Domineering. Section 2 gives an overview of related research. In Section 3 we describe

our methods implemented. Next, in Section 4, we show three series of experiments, one on the generation and statistical analysis of the databases up to size 15, a second on the impact of several CGT enhancements on solving power using all databases, and the third one on the effect of database size for a configuration with most enhancements enabled. In Section 5 we give our conclusions and some suggestions for further research. Note that we refrain from giving an introduction to CGT, due to space restrictions. For this we refer to the literature mentioned above, in particular to [6, 3, 1].

2 Related Work

Some related work has been done by Müller [11] who applied CGT values to solve local endgames in Go. However, the global search is not an α-β search, and CGT values are not obtained from CGT endgame databases, but calculated on the spot. Müller and Li [13] show results for combining α-β search with CGT pruning and ordering. Their games are artificial games with very special properties, having nothing in common with "real" combinatorial games. No endgame databases were used.

Most other related work has been done in the area of the game Amazons. Müller [12] used CGT to establish bounds in a specialized divide-and-conquer approach to solve Amazons boards. He was able to solve 5×5 Amazons. No CGT endgame databases were used. Snatzke [16] built CGT endgame databases for a very restricted version of Amazons, namely for subgames fitting in a 2×11 board with exactly 1 queen per player. This was extended in [17] with new results for some small databases of other shapes, with 1 to 4 queens. He did not incorporate the use of his databases in a general Amazons solver. Tegos [19] was the first to combine endgame databases for Amazons in a NegaScout-based Amazons playing program. Besides (traditional) minimax endgame databases (without CGT information) he also implemented CGT endgame databases. These contained just thermograph information, not exact CGT values, and therefore only could be used for (heuristic) move-ordering purposes. Recently, Song [18] implemented endgame databases in an Amazons solver. Again, the databases did not contain exact CGT values, but heuristic (thermograph) information useful for narrowing the bounds in the solving process. With his program 5×6 Amazons has been solved.

As far as we know there is no other literature reporting on combining global α-β searches with endgame databases with precise CGT values. So, to the best of our knowledge our implementation is the first of such a kind.

3 Methods

In this section we describe the methods implemented in our Domineering solver. This consists of the construction of the CGT endgame databases (Section 3.1), the basic NegaScout solver (Section 3.2), and several ways how the CGT values from the endgame databases can be used within the α-β framework (Section 3.3).

3.1 CGT Databases

In order to use CGT values for Domineering subgames we have built all Domineering endgame databases for subgames of sizes 1–15. Positions that exhibit mirror symmetry (left/right and/or top/bottom) are unified into a single entry. No 90° rotation symmetry, i.e. switching the players, is applied at present. The CGT values for the Domineering subgames are calculated with the CGSUITE software tool [15]. These values are in so-called canonical form, which make them a unique representation of the values.

3.2 Basic α-β solver

For the basic Domineering solver we built a NegaScout [14] implementation of a very simple α-β searcher. This searcher investigates lines until the end (returning that either Vertical or Horizontal made the last move and so wins the game). Besides this, the only enhancement is that the moves of a player are ordered descendingly by the difference in the current player's and opponent's sums of safe and real moves. This ordering has been used before by Breuker *et al.* to solve 8×8 Domineering [4]. Since our goal was to measure the impact of using CGT on solution tree sizes, we refrained from incorporating any further enhancements to the α-β framework.

3.3 Combining CGT Endgame Databases with α-β

In this subsection we describe several enhancements for combining CGT with the α-β solver. To be able to easily assess their individual effects, we have implemented these as options, that can independently be turned on or off.

The Subgame Ordering option
First we have an ordering heuristic for investigating the moves. Here, if we have more than one disjoint subgame, we first investigate only moves within subgames of unknown value. Within all unknown subgames together the moves are ordered using the standard move ordering, ties being broken in favour of the smallest unknown subgame. Next, moves in hot subgames in order of decreasing temperature (see, e.g., [1]) are investigated. Then the infinitesimal subgames are investigated. Since they all have temperature 0, we order them descendingly in their absolute distance to 0, as determined by CGT. Finally, the pure number subgames are investigated in ascending order of their absolute values.

The Prune Numbers option
When the Domineering board has been decomposed into disjoint subgames, all with values being numbers, there is no need to play any further. The result is completely determined by the sign of the sum N of the numbers, being a Vertical win if $N > 0$, a Horizontal win if $N < 0$, and a 2^{nd}-player win if $N = 0$. The last case means that the winner is the one who made the last move.

The Prune Switches option
When the Domineering board has been decomposed into disjoint subgames, all with values being numbers or switches, the determination is also straightforward. Switches are distributed between players in order of descending temperature. The final value determines the winner as given above.

The Prune Infinitesimals option
In case the sum of the number and/or switch subgames equals zero, we still can determine the outcome of a subgame when the remaining subgames are pure infinitesimals. For any combination of infinitesimal subgames consisting of stars, ups and downs, and tinies and minies, CGT enables determining the winner of the sum of subgames.

The Prune all option
When the Domineering board has been decomposed into disjoint subgames, all with values being any combination of numbers, switches, and infinitesimals, CGT enables determining the winner of the sum of subgames.

The Split Subgames option
Whereas all previous options are valid for every type of combinatorial game, there is a famous theorem specifically valid for the game of Domineering, first given by Conway [6], that gives a useful way to split a subgame into two smaller subgames.

Theorem 1 (The splitting theorem) *If* $G\square = G$ *then* $G\square H = G + \square H$

This theorem says that if a subgame consists of two parts, say G and H, separated by a single bridging square, such that G and H are attached to two opposite sides of the bridging square and G and H have no edges in common, and it further holds that the CGT value of G alone equals the CGT value of G plus the bridging square (but without H), then the subgame can be split into two separate subgames G and (H with the bridging square). Note that this splitting may be applied repeatedly, but only sequentially, not simultaneously.

4 Experiments and Discussion

In this section we describe experimental results. Our test set consists of all rectangular boards with dimensions from 2 to 7. Using this test set, we reveal the impact of using CGT endgame databases on the size of the solution trees for solving the 36 boards. Section 4.1 gives some statistics on the databases generated (up to size 15). Section 4.2 shows the impact for the largest database generated (size 15) on solving the test positions, for various options for using CGT values. The last subsection, Section 4.3, shows what the effect of database size is on solving power.

4.1 Subgame Databases

We have generated the subgame databases up to a maximum position size of 15 tiles. The maximum subgame size is hereafter also referred to as database size. Beyond that point, the number of entries quickly grows too large to fit in the average home computer's main memory. A survey of the completed databases shows that the number of entries approximately increases by a factor of between 3 and 4 with every additional subgame tile. At the final size of 15, the databases contain a total of about 9.3 million entries.

Table 1 gives an overview of how positions of a fixed size are distributed over the various types of game theoretical values. Generally, each column describes one distinct type of value or a subset of them. The first value column lists the numbers of games that are zero, i.e., wins for the previous player. The second column accounts for all remaining, non-zero number games. The third column gives the numbers of games that are non-zero multiples of ups and downs, i.e., $n \cdot \uparrow$ and $n \cdot \downarrow$ with $n > 0$. The next column represents all nimber games except $*0$ (since $*0$ is equivalent to the number 0 and thus already given in the counts of the first value column). The column $+_x$ / $-_x$ contains the numbers of games that have value tiny-x and miny-x, with $x > 0$. To account for combinations of these basic values, the column Comb. counts all occurrences of the previous types and any combinations thereof. The last two columns list the numbers of switch games and the total numbers of games of a particular position size.

Pos. Size	Num. $= 0$	Num. $\neq 0$	$n \cdot \uparrow$ $n \cdot \downarrow$	$*n$ $\neq *0$	$+_x$ $-_x$	Comb.	Switch	Total
2	0	2	0	0	0	2	0	2
3	0	2	0	1	0	3	0	3
4	0	4	0	2	0	6	3	9
5	5	4	0	1	0	16	5	21
6	10	16	2	0	0	38	18	68
7	13	48	4	13	0	106	54	208
8	16	194	6	46	4	320	183	730
9	116	386	2	104	6	950	572	2542
10	515	1262	94	136	8	3125	2134	9287
11	1061	3570	336	462	0	9867	6797	34,053
12	2074	14,700	764	3618	28	32,990	24,760	127,112
13	5012	45,018	1392	13,768	188	108,994	77,941	476,849
14	27,816	155,410	5018	24,002	820	367,330	277,045	1,803,636
15	135,539	437,718	23,752	46,254	1988	1,237,853	893,180	6,851,960
TOTAL	172,177	658,334	31,370	88,407	3042	1,761,600	1,282,692	9,306,480

Table 1: Distribution of Domineering positions of a fixed size over the different types of game theoretical values.

Clearly, as the overall number of entries grows with increasing position size, the ratio of all simple value types gradually decreases. Even at the maximum size, however, still approximately 8% of the entries are number games and 13% are switch games. When considering that they are usually the most decisive values during subgame pruning, it is fortunate that they make up a noteworthy portion of the entries.

4.2 Impact of CGT Enhancements

For the next series of experiments we use the subgame database of size 15. In the course of the experiments the different CGT enhancements are gradually enabled to compare their effectiveness in terms of numbers of explored game positions. Table 2 presents the results for all Domineering game boards in the test set. Vertical always starts. The second column denotes the winner, V for Vertical, H for Horizontal. These results fully comply with previous results [4, 5]. The first results column provides a baseline using only the standard alpha-beta move ordering. Over the next six columns, one enhancement at a time is enabled in addition to the previous configuration: subgame ordering, subgame pruning on numbers, subgame pruning on switches, subgame pruning on infinitesimals, subgame pruning on any combination of numbers, switches and infinitesimals, and subgame splitting.

Board Size	W	Std. Move Ordering	+ Subgame Ordering	+ Pruning Numbers	+ Pruning Switches	+ Pruning Infinits	+ Pruning All	+ Subgame Splitting
2×2	V	2	2	2	1	1	1	1
2×3	V	2	2	2	1	1	1	1
2×4	H	9	9	5	3	1	1	1
2×5	V	6	6	1	1	1	1	1
2×6	V	12	12	5	4	2	1	1
2×7	V	14	14	2	1	1	1	1
3×2	V	4	4	2	1	1	1	1
3×3	V	6	6	2	1	1	1	1
3×4	H	13	13	1	1	1	1	1
3×5	H	61	61	1	1	1	1	1
3×6	H	125	119	35	7	7	7	5
3×7	H	315	319	53	33	33	33	24
4×2	V	10	10	4	3	1	1	1
4×3	V	18	18	1	1	1	1	1
4×4	V	192	192	68	34	6	4	4
4×5	V	448	436	133	26	26	25	16
4×6	V	11,448	11,216	2899	1795	896	496	157
4×7	V	27,102	25,294	3850	1606	1280	778	535
5×2	H	13	13	1	1	1	1	1
5×3	V	26	26	1	1	1	1	1
5×4	H	577	561	128	82	59	37	19
5×5	H	7973	7385	1021	493	461	405	273
5×6	H	51,181	48,663	8960	5323	4313	3970	1660
5×7	H	746,077	697,739	71,242	54,175	45,269	43,028	20,018
6×2	V	18	18	10	6	6	1	1
6×3	V	50	50	18	10	10	10	10
6×4	V	3232	3174	810	410	244	134	58
6×5	V	27,812	25,120	4577	3372	2819	2557	1049
6×6	V	1,283,694	1,189,800	195,495	127,084	67,997	38,633	17,716
6×7	V	24,555,340	22,141,400	2,405,687	1,603,534	1,154,749	979,767	399,859
7×2	V	40	38	17	1	1	1	1
7×3	V	210	210	36	20	20	20	18
7×4	H	51,105	48,487	7137	3635	2482	1778	1187
7×5	V	251,364	238,288	37,594	23,695	21,132	20,860	9169
7×6	H	77,980,559	67,623,723	5,392,282	3,726,686	2,474,996	2,187,598	963,222
7×7	V	1,487,411,406	1,299,871,506	70,706,836	52,922,023	40,960,753	39,777,405	13,331,919
TOTAL		1,592,410,464	1,391,933,934	78,838,918	58,474,071	44,737,574	43,057,561	14,746,934

Table 2: The numbers of explored Domineering positions in several series of experiments using the database with a maximum subgame size of 15, and gradually activating the various CGT search enhancements.

The first CGT enhancement, subgame ordering, shows a small but significant reduction of 12.6% on average. The number of explored positions remains unchanged or decreases for 35 out of 36 experiments.

With the activation of the pruning options we get large average reductions compared to the base line experiment. When activating the numbers pruning option, the two experiments on the very smallest game boards 2×2 and 2×3 (which are switches) attain the same results as before, and all other experiments experience reduction rates between 44 and 98%, with an average reduction rate of 95.0%. Activating switches pruning, infinitesimals pruning and all pruning reduces this further via 96.3% and 97.2% to 97.3% average reduction.

The final experiment that uses all CGT enhancements achieves an average improvement of 99.1% as compared to the baseline experiment. So, using this configuration, the test set is solved in less than 1% of the original number of explored positions, which means that more than 99% of all work can be skipped.

We also observe that the results confirm the function of the implemented subgame pruning. All game boards with a number of tiles ≤ 15 can be solved in a single exploration step.

4.3 Impact of Database Size

The final series of experiments investigates the influence of the maximum subgame size of the databases on the efficiency of CGT-assisted alpha-beta pruning. The complete set of CGT search enhancements except the all pruning and subgame splitting are used, and the databases are gradually enlarged from sizes 4 to 15 in steps of 1. Subgame splitting is excluded from these experiments because by breaking large games into smaller components, the larger databases' potential may not be fully utilized.

The results are displayed in Table 3. The numbers confirm that with every increment in database size, in the most general sense, the search performance improves. Some transitions between databases appear to be more effective than others, but overall a larger database size yields better results.

It is quite difficult to make out at first glance, but in addition to this rather anticipated result of gradual improvement, there is also a remarkable pattern at work here. This pattern does not always begin to show immediately because the smallest databases do not yet contain all major subgames. Once the most important positions are available, however, the relative performance improvement of a database over the next smaller database fluctuates. When for example analyzing the board size 7×7, the relative improvement of switching between database sizes 10 to 11 is 19%. Switching from 11 to 12, from 12 to 13, from 13 to 14, and from 14 to 15 yield relative improvements of 13%, 17%, 11%, and 16%, respectively. In this example, clearly, utilizing databases with an odd maximum subgame size results in a larger performance boost than an even maximum subgame size. In other cases, on the contrary, switching to an even database size proves more beneficial than the alternative.

The pattern that is embedded here is based on the game board size, and specifically on the total number of tiles. All boards that have an even number of tiles benefit more from switching to databases of even size than switching to an odd size. All boards that have an odd number of tiles behave exactly opposite to that. The reason for this most likely has to do with the nature of moves in the game of Domineering; that is the 2×1 shape of the game pieces. As a piece always occupies two tiles of the game board, on a board with an even number of tiles it can be possible to fill the entire board. On the other hand, a board with an odd number of tiles can never be completely filled because there are no half-pieces. In general, this suggests that boards with even numbers of tiles decompose more easily into subgames of even size than boards with odd numbers of tiles do. Analogously, boards with odd numbers of tiles appear to decompose more frequently into subgames of odd size than boards with even numbers of tiles do. This probably explains why different board sizes benefit more from certain database sizes than from others.

5 Conclusions and Future Research

We have shown how CGT values of subgames can be used in several ways as refinements of a basic NegaScout solver. Although the subgame ordering yields only a limited reduction in the number of positions explored, all other CGT enhancements yield large reductions in the sizes of the solution trees built. When all CGT options are enabled reductions up to 99 % are achieved.

As future research we see several lines for continuation. First, we intend to enhance the α-β implementation in several ways, such as incorporating transposition tables and better move ordering heuristics. We are convinced that with a full-fledged α-β solver incorporating the CGT endgame databases we will be able to solve more Domineering boards than before, surpasssing the current state-of-the-art Domineering solver OBSEQUI [5]. We also want to do more experiments on the dependencies of the different CGT options among themselves and with other move ordering heuristics, and on the effect of the database size, especially with the all-pruning and subgame-splitting options enabled. In addition we then also hope to get more insight into the odd-even effect mentioned in Section 4.3.

Secondly, another line of current research concentrates on so-called *perfectly solving* Domineering boards [20], i.e., solving without any search at all. This is a very knowledge-intensive approach. We believe that the proper combination of this knowledge-based approach with the efficiency of a strong α-β solver with CGT endgame databases may advance the solving power for Domineering even more.

Thirdly, in order to see whether the good performance of CGT endgame databases is typical for Domineering or also extends to other combinatorial games, we intend to build endgame databases and incorporate them also into α-β solvers for other games, such as Clobber and Amazons.

Board Size	4	5	6	7	8	9	10	11	12	13	14	15
2×2	1	1	1	1	1	1	1	1	1	1	1	1
2×3	2	2	1	1	1	1	1	1	1	1	1	1
2×4	4	4	3	3	1	1	1	1	1	1	1	1
2×5	3	3	2	2	2	2	1	1	1	1	1	1
2×6	6	6	4	4	2	2	2	2	2	2	2	2
2×7	6	6	6	6	4	4	2	2	2	2	1	1
3×2	2	2	1	1	1	1	1	1	1	1	1	1
3×3	4	4	4	2	2	1	1	1	1	1	1	1
3×4	9	9	7	7	5	5	3	3	1	1	1	1
3×5	37	31	28	23	22	15	15	7	7	4	4	1
3×6	57	54	47	46	41	40	33	32	19	19	7	7
3×7	203	187	157	136	122	108	99	77	74	58	57	33
4×2	6	6	3	3	1	1	1	1	1	1	1	1
4×3	10	10	8	8	6	6	2	2	1	1	1	1
4×4	91	78	53	53	41	41	29	29	14	14	6	6
4×5	182	174	160	153	117	112	83	83	54	54	26	26
4×6	4911	3880	3220	2618	2166	1843	1478	1411	1136	1110	914	896
4×7	8468	7035	5321	4916	3809	3592	3023	2878	2198	2140	1281	1280
5×2	5	5	5	5	3	3	1	1	1	1	1	1
5×3	14	14	14	14	14	12	12	8	8	2	2	1
5×4	328	267	219	204	156	147	104	96	70	70	59	59
5×5	3664	2551	2197	1740	1606	1358	1232	1027	952	766	724	461
5×6	31,401	23,926	17,319	15,172	12,437	10,877	9093	8036	6604	6001	4582	4313
5×7	426,730	288,380	218,651	169,663	134,307	110,807	96,931	80,550	71,472	59,406	54,140	45,269
6×2	10	10	7	7	6	6	6	6	6	6	6	6
6×3	48	42	34	34	30	30	22	22	16	16	10	10
6×4	1273	1072	953	756	584	578	374	362	285	285	244	244
6×5	15,054	12,735	9005	7804	5989	5553	4554	4241	3602	3379	2907	2819
6×6	938,471	749,470	224,925	183,648	147,759	130,166	106,666	98,273	82,817	79,260	69,410	67,997
6×7	37,709,960	27,474,162	4,988,723	4,022,705	3,099,763	2,721,756	2,253,398	2,007,299	1,664,639	1,501,653	1,255,777	1,154,749
7×2	18	18	12	12	8	8	8	8	8	8	1	1
7×3	158	134	127	110	106	88	86	62	58	46	46	20
7×4	19,968	16,109	12,344	10,622	7925	7015	5358	4844	3534	3216	2575	2482
7×5	182,720	128,671	96,596	74,615	60,656	51,984	45,957	39,219	34,577	29,311	26,897	21,132
7×6	55,495,123	34,416,243	13,329,538	10,243,866	7,792,186	6,579,031	5,340,985	4,630,721	3,748,753	3,310,783	2,734,423	2,474,996
7×7	3,912,866,220	949,064,598	517,180,959	245,694,475	154,497,361	115,204,980	97,508,171	78,409,226	67,567,401	55,563,969	48,978,827	40,960,753

Table 3: The numbers of explored Domineering positions in several series of experiments using all CGT search enhancements except the all prune option and subgame splitting, and gradually incrementing the databases' maximum subgame size from 4 to 15.

References

[1] M.H. Albert, R.J. Nowakowski, and D. Wolfe. *Lessons in Play: An Introduction to Combinatorial Game Theory*. A K Peters, Wellesley, MA, 2007.

[2] E.R. Berlekamp. Blockbusting and Domineering. *J. Combin. Theory* (Ser. A), 49:67–116, 1988.

[3] E.R. Berlekamp, J.H. Conway, and R.K. Guy. *Winning Ways for your Mathematical Plays*, volume 1-2. Academic Press, London, 1982. 2nd edition, in four volumes: vol. 1 (2001), vols. 2, 3 (2003), vol. 4 (2004). A K Peters, Wellesley, MA.

[4] D. M. Breuker, J.W.H.M. Uiterwijk, and H.J. van den Herik. Solving 8×8 Domineering. *Theoret. Comput. Sci.* (Math. Games), 230:195–206, 2000.

[5] N. Bullock. Domineering: Solving large combinatorial search spaces. *ICGA Journal*, 25:67–84, 2002.

[6] J.H. Conway. *On Numbers and Games*. Academic Press, London, 1976. 2nd edition (2001): A K Peters, Natick, MA.

[7] M. Gardner. Mathematical games. *Scientific American*, 230:106–108, 1974.

[8] H.J. van den Herik, J.W.H.M. Uiterwijk, and J. van Rijswijck. Games solved: Now and in the future. *Artificial Intelligence*, 134:277–311, 2002.

[9] Y. Kim. New values in Domineering. *Theoret. Comput. Sci.* (Math. Games), 156:263–280, 1996.

[10] D.E. Knuth and R.W. Moore. An analysis of alpha-beta pruning. *Artificial Intelligence*, 6:293–326, 1975.

[11] M. Müller. Global and local game tree search. *Information Sciences*, 135:187–206, 2001.

[12] M. Müller. Solving 5x5 Amazons. In *The 6th Game Programming Workshop (GPW 2001), Hakone (Japan), 2001*, volume 14 of *IPSJ Symposium Series*, pages 64–71, 2001.

[13] M. Müller and Z. Li. Locally informed global search for sums of combinatorial games. In H.J. van den Herik, Y. Björnsson, and N. Netanyahu, editors, *Computers and Games: 4th International Conference, CG 2004*, volume 3846 of *Lecture Notes in Computer Science*, pages 273–284. Springer Int. Publ., 2006.

[14] A. Reinefeld. *Spielbaum-Suchverfahren*. Informatik-Fachberichte. Springer, 1989.

[15] A.N. Siegel. Combinatorial game suite: A computer algebra system for research in combinatorial game theory, 2003. Available from `http://cgsuite.sourceforge.net/`.

[16] R.G. Snatzke. Exhaustive search in the game Amazons. In R.J. Nowakowski, editor, *More Games of No Chance,* Proc. MSRI Workshop on Combinatorial Games, July, 2000, Berkeley, CA, MSRI Publ., volume 42, pages 261–278. Cambridge University Press, Cambridge, 2002.

[17] R.G. Snatzke. New results of exhaustive search in the game Amazons. *Theoret. Comput. Sci.*, 313:499–509, 2004.

[18] J. Song. An enhanced solver for the game of Amazons. Master's thesis, University of Alberta, 2013.

[19] T. Tegos. Shooting the last arrow. Master's thesis, University of Alberta, 2002.

[20] J.W.H.M. Uiterwijk. Perfectly solving Domineering boards. In T. Cazenave, M.H.M. Winands, and H. Iida, editors, *Workshop on Computer Games, CGW 2013 at IJCAI 2013*, volume 408 of *Communications on Computer and Information Science*, pages 97–121. Springer Int. Publ., 2014.

[21] J.W.H.M. Uiterwijk and H.J. van den Herik. The advantage of the initiative. *Information Sciences*, 122:43–58, 2000.

[22] D. Wolfe. Snakes in Domineering games. *Theoret. Comput. Sci.* (Math. Games), 119:323–329, 1993.

Using Distances for Aggregation in Abstract Argumentation

Richard Booth [1] Mikołaj Podlaszewski [1]

CSC / SnT, University of Luxembourg
{richard.booth,mikolaj.podlaszewski}@uni.lu

Abstract

We continue recent work on the problem of aggregating labellings of an argumentation framework by adapting the *distance-based* framework of Miller and Osherson from binary judgment aggregation to the argumentation setting. To instantiate the framework we employ some notions of *labelling-distance* recently introduced by Booth et al., in the process generalising and extending some of the latters' results. We introduce some new postulates for distance methods based on the concept of *qualitative distance*, and check their validity.

1 Introduction

The aggregation of conflicting opinions into a collective one is fundamental in multi-agent systems. Individuals presented with the same set of conflicting arguments might take different rational positions. The problem of aggregation in abstract argumentation has been explored in a number of recent papers [1, 4, 10] which employ techniques from *judgment aggregation* (JA) [8] to the problem of aggregating 3-valued *argument labellings*. These works have shown that, as with classical JA, it is not possible to define general aggregation operators that satisfy a number of seemingly mild constraints while ensuring collective rationality of the outcome.

One way of getting around this problem in JA is to use one of the *distance-based solution methods* that were studied by Miller and Osherson (hereafter MO)[9] within the framework of binary judgment aggregation. As the name suggests these depend on a provided notion of *distance measure* between binary judgment sets. In this paper we show how this option can also be used in the argumentation setting. We first modify the MO framework for our purposes, before bringing in some notions of distance between argument labellings that have been defined in [2]. We thus illustrate the usefulness of the distance measures defined in that paper.

Along the way we generalise and extend some of the results in [2]. For example some of the MO aggregation methods require a distance to be defined between any two arbitrary labellings of an argumentation framework, whereas the most interesting distance measures of [2], such as the *issue-based* distance, are defined only between *complete* labellings. We thus extend the definition of these distances to apply to arbitrary labellings. We also look at two new postulates for distance measures between labellings, based on the *Normality* and *Increasing* properties introduced by MO in the JA setting, and confirm to what extent our distance methods conform with these postulates.

As well as a distance-measure, the MO methods also require up-front specification of an *initial* aggregation operator, which intuitively can be thought of as a *gold standard* operator that satisfies a number of basic postulates, without always yielding collectively rational results. In fact the MO methods can be viewed as offering recipes to *repair* the result of this operator in the cases when it does not give a collectively rational outcome. In the argumentation setting, Caminada and Pigozzi already suggested another way to carry out such a repair in these cases, using what they called the *down-admissible* and *up-complete* procedures [4]. Our work thus provides an alternative solution.

[1] Supported by the National Research Fund, Luxembourg (DYNGBaT and LAAMIcomp projects)

2 Preliminaries

We use the familiar setting of *abstract argumentation* [5]. We start by assuming a countably infinite set U of argument names, from which all possible argumentation frameworks are built. We restrict ourselves to finite argumentation frameworks.

Definition 1. *An* argumentation framework *(AF for short)* $\mathscr{A} = (Args, \rightharpoonup)$ *is a pair consisting of a finite set* $Args \subseteq U$ *of arguments and an attack relation* $\rightharpoonup \subseteq Args \times Args$. *We also use* $Args_{\mathscr{A}}$ *and* $\rightharpoonup_{\mathscr{A}}$ *to denote the arguments and attack relation of a given AF* $\mathscr{A}$.

Definition 2. *Let* $\mathscr{A} = (Args, \rightharpoonup)$ *be an AF. An* $\mathscr{A}$-labelling *is a function* $L : Args \to \{\texttt{in}, \texttt{out}, \texttt{undec}\}$. *The set of all* $\mathscr{A}$*-labellings is denoted by* $Labs(\mathscr{A})$. *Given* $A \subseteq Args$ *we denote by* $L[A]$ *the restriction of* L *to* A.

For notational purposes it is useful to define a unary "negation" operator on the set of labels by $\neg\texttt{in} = \texttt{out}$, $\neg\texttt{out} = \texttt{in}$ and $\neg\texttt{undec} = \texttt{undec}$.

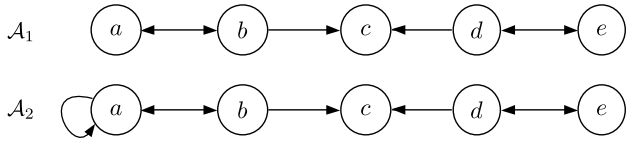
Of course a *rational* labelling should somehow respect the attack relation. This is embodied in the following definition:

Definition 3. *Let* $\mathscr{A}$ *be an AF and* $L \in Labs(\mathscr{A})$. *For any argument* $a \in Args_{\mathscr{A}}$ *we say:*

- *a is legally* `in` *if* $L(a) = \texttt{in}$ *and* $L(b) = \texttt{out}$ *for all* $b \in Args_{\mathscr{A}}$ *s.t.* $b \rightharpoonup_{\mathscr{A}} a$,
- *a is legally* `out` *if* $L(a) = \texttt{out}$ *and* $L(b) = \texttt{in}$ *for some* $b \in Args_{\mathscr{A}}$ *s.t.* $b \rightharpoonup_{\mathscr{A}} a$,
- *a is legally* `undec` *if* $L(a) = \texttt{undec}$ *and there is no* $b \in Args_{\mathscr{A}}$ *s.t.* $b \rightharpoonup_{\mathscr{A}} a$ *and* $L(b) = \texttt{in}$, *and there exists* $c \in Args_{\mathscr{A}}$ *s.t.* $c \rightharpoonup_{\mathscr{A}} a$ *and* $L(c) = \texttt{undec}$.

L is complete *iff it has no illegally* `in` *and no illegally* `out` *and no illegally* `undec` *arguments. We denote the set of complete* $\mathscr{A}$*-labellings by* $Comp(\mathscr{A})$.

In the rest of this paper we identify rational $\mathscr{A}$-labellings with complete $\mathscr{A}$-labellings. This is because they form the basis for other semantics such as preferred, stable, semi-stable, etc (see [3]). This choice is also in line with other works on aggregation in argumentation [1, 4, 10]. However we will also make use later of the notion of *admissible* $\mathscr{A}$-labelling, which is an $\mathscr{A}$-labelling containing no illegally `in` or illegally `out` arguments (but possibly containing illegally `undec` arguments).



	l_1	l_2	l_3	l_4	l_5	l_6	l_7	l_8	l_9	n_1	n_2	n_3
a	o	o	o	u	u	u	i	i	i	u	i	u
b	i	i	i	u	u	u	o	o	o	u	o	u
c	o	o	o	o	u	u	o	i	u	o	u	i
d	i	o	u	i	o	u	i	o	u	u	o	o
e	o	i	u	o	i	u	o	i	u	u	i	i

Figure 1: Two aggregation frameworks with all their complete labellings ($\mathscr{A}_1$: l_1-l_9, $\mathscr{A}_2$: l_1-l_6), and three other labellings (n_1-n_3).

Example 1. *AFs can be visualised as directed graphs with nodes and edges representing arguments and attacks respectively. Throughout the paper we will use the following two running example AFs* $\mathscr{A}_1$ *and* $\mathscr{A}_2$, *both consisting of the five arguments* $\{a,b,c,d,e\}$. *We represent labellings by a string of letters* `i`, `u` *and* `o` *corresponding to* `in`, `undec` *and* `out` *respectively. AF* $\mathscr{A}_1$ *has 9 complete labellings* l_1-l_9. *In* $\mathscr{A}_2$ *labellings* $l_7 - l_9$ *are no longer complete because argument a is illegally-*`in` *due to the additional attack. For both AFs the labellings* n_1 *and* n_3 *are not admissible (and hence not complete). For* $\mathscr{A}_1$ *the labelling* n_2 *is admissible but not complete. For* $\mathscr{A}_2$ *it is even not admissible because argument a is illegally-*`in`.

Now we come to aggregation. We assume a set of agents $Ag = \{1, \ldots, n\}$ (with $n \geq 2$) is fixed.

Definition 4. *Let* $\mathscr{A}$ *be an AF. An* $\mathscr{A}$-profile *is any n-tuple of* $\mathscr{A}$*-labellings* $\mathbf{L} = (L_1, \ldots, L_n)$. *If every* L_i *is a complete* $\mathscr{A}$*-labelling then we call* $\mathbf{L}$ *a* complete $\mathscr{A}$*-profile.*

What we seek is a way to construct aggregation operators that, given any $\mathscr{A}$-profile $\mathbf{L}$ as input, return a set of $\mathscr{A}$-labellings $F_{\mathscr{A}}(\mathbf{L})$. Note that an aggregation operator is always defined in a context of some specific AF $\mathscr{A}$. More generally we are interested in an aggregation *method* that given any context AF $\mathscr{A}$ will return in a systematic way an aggregation operator for $\mathscr{A}$.

Definition 5. *Let $\mathscr{A}$ be an AF. An* (irresolute) aggregation operator *(for $\mathscr{A}$) is a function $F_{\mathscr{A}}$ that assigns, to each $\mathscr{A}$-profile* $\mathbf{L}$ *a set $F_{\mathscr{A}}(\mathbf{L}) \subseteq Labs(\mathscr{A})$. An* aggregation method *is a mapping that, given any context AF $\mathscr{A}$, returns an aggregation operator $F_{\mathscr{A}}$ for $\mathscr{A}$.*

From now on we will drop the *irresolute* and just say *aggregation operator*. In previous works on aggregation in argumentation the output is usually taken to be a single labelling, but we relax that here. When important, we say a *resolute* aggregation operator for an operator which returns always a singleton set. Also note for each $\mathscr{A}$, $F_{\mathscr{A}}$ is defined for *all* $\mathscr{A}$-profiles (not necessarily just the complete ones), and that the output of $F_{\mathscr{A}}(\mathbf{L})$ is allowed to be *any* subset of $\mathscr{A}$-labellings. Ideally, of course, we would like the output to consist only of *complete* $\mathscr{A}$-labellings, i.e., we want the following to hold:

Collective Rationality For all $\mathscr{A}$ and $\mathscr{A}$-profiles $\mathbf{L}$, $F_{\mathscr{A}}(\mathbf{L}) \subseteq Comp(\mathscr{A})$

3 Miller and Osherson aggregation methods.

Miller and Osherson (hereafter MO) [9] described a framework for using *distance measures* to define aggregation methods in binary judgment aggregation. In that setting agents evaluate a set of logical propositions called an *agenda* by providing a *judgement set* which is an assignment of either True or False to each proposition in the agenda. Their framework requires specification of two things:

1. An *initial* ***resolute*** *aggregation method M* that is able to give collectively rational answers in simple cases. In their case they only considered the proposition-wise majority method, but in principle any M can be used. The intuition is that if the outcome produced by M happens to be collectively rational then there is no need to choose a different outcome.

2. A measure of *distance* between any two judgment sets of any given agenda. This measure is assumed to be a *metric*.

Definition 6. *A* metric *on a set X is a function $d : X \times X \to \mathbb{R}$ such that for all $x, y, z \in X$,* (i) *if $d(x,y) = 0$ then $x = y$,* (ii) *$d(x,x) = 0$,* (iii) *$d(x,y) = d(y,x)$,* (iv) *$d(x,z) \leq d(x,y) + d(y,z)$. A function d satisfying* (ii)-(iv) *but not* (i) *is called a* pseudometric.

In our argumentation setting, the roles of the agenda and the judgment set are filled by the AF $\mathscr{A}$ and $\mathscr{A}$-labelling respectively. We will make use of a *distance method*, which for each AF $\mathscr{A}$ returns a distance measure over $\mathscr{A}$-labellings.

Definition 7. *Let $\mathscr{A}$ be an AF. A* distance measure *$d_{\mathscr{A}}$* for *$\mathscr{A}$ is a metric on the set $Labs(\mathscr{A})$. A* distance method *d is a function that associates a distance measure $d_{\mathscr{A}}$ to each AF $\mathscr{A}$.*

We can extend a distance measure $d_{\mathscr{A}}$ so that it also returns distance from an $\mathscr{A}$-profile to an $\mathscr{A}$-labelling, as well as distance between 2 $\mathscr{A}$-profiles. For $L \in Labs(\mathscr{A})$ and profiles $\mathbf{L} = (L_1, \ldots, L_n)$, $\mathbf{L}' = (L'_1, \ldots, L'_n) \in Labs(\mathscr{A})^n$ we define

$$d_{\mathscr{A}}(\mathbf{L}, L) = \sum_{i=1}^{n} d_{\mathscr{A}}(L_i, L), \qquad d_{\mathscr{A}}(\mathbf{L}, \mathbf{L}') = \sum_{i=1}^{n} d_{\mathscr{A}}(L_i, L'_i).$$

Also required for MO is the notion of *M-consistent* profile. These are the profiles that, when passed to M, result in a collectively rational outcome.

Definition 8. *Let $\mathscr{A}$ be an AF, $\mathbf{L} \in Labs(\mathscr{A})^n$ and M a resolute aggregation method. Then $\mathbf{L}$ is M-*consistent (for $\mathscr{A}$) *iff $M_{\mathscr{A}}(\mathbf{L}) \in Comp(\mathscr{A})$. We denote by $Cons_{\mathscr{A}}(M)$ the set of M-consistent $\mathscr{A}$-profiles, and by $Cons_{\mathscr{A}}(M, Comp)$ the set $Cons_{\mathscr{A}}(M) \cap Comp(\mathscr{A})^n$.*

MO describe four different ways in which all the above ingredients can be combined, resulting in four classes of aggregation methods which we now describe. First some notation: For any function $f : X \to Y$ and sets $D \subseteq X$, $C \subseteq Y$ we denote the image of D by $f(D) = \{f(x) \mid x \in D\}$ and the inverse image of C by $f^{-1}(C) = \{x \in X \mid f(x) \in C\}$. The subset of D for which f obtains its minimal value is returned by the operator $\arg\min_{x \in D} f(x) = \{x \in D \mid f(x) \leq f(x') \text{ for all } x' \in D\}$.

Definition 9 ([9]). *Let d be distance method and M a resolute aggregation method. The four aggregation methods $\mathtt{P}^d$, $\mathtt{E}^{M,d}$, $\mathtt{F}^{M,d}$ and $\mathtt{O}^{M,d}$ are defined by setting, for each AF $\mathscr{A}$ and $\mathscr{A}$-profile $\mathbf{L}$:*

$$\mathtt{P}^d_{\mathscr{A}}(\mathbf{L}) = \operatorname*{arg\,min}_{L \in Comp(\mathscr{A})} d_{\mathscr{A}}(\mathbf{L}, L) \qquad \mathtt{F}^{M,d}_{\mathscr{A}}(\mathbf{L}) = M_{\mathscr{A}}\left(\operatorname*{arg\,min}_{\mathbf{L}' \in Cons_{\mathscr{A}}(M,Comp)} d_{\mathscr{A}}(\mathbf{L}, \mathbf{L}')\right)$$

$$\mathtt{E}^{M,d}_{\mathscr{A}}(\mathbf{L}) = \operatorname*{arg\,min}_{L \in Comp(\mathscr{A})} d_{\mathscr{A}}(M_{\mathscr{A}}(\mathbf{L}), L) \qquad \mathtt{O}^{M,d}_{\mathscr{A}}(\mathbf{L}) = M_{\mathscr{A}}\left(\operatorname*{arg\,min}_{\mathbf{L}' \in Cons_{\mathscr{A}}(M)} d_{\mathscr{A}}(\mathbf{L}, \mathbf{L}')\right)$$

All four MO aggregation methods minimise distance to ensure the collective outcome is rational. The P and E methods minimise the distance over all complete labellings. P (or Prototype) returns the complete labellings closest to the profile $\mathbf{L}$. E (or Endpoint) returns the complete labellings closest to the labelling returned by initial aggregator $M_{\mathscr{A}}(\mathbf{L})$, which possibly is not complete. The F and O (for Full and Output resp.) methods select the M-consistent profiles closest to $\mathbf{L}$ and then applies $M_{\mathscr{A}}$ to them. The difference between these two is that O performs its selection from among *all* M-consistent profiles, while F selects only from those that are, in addition, themselves complete.

Some observations in these definitions: *(i)* All four aggregation methods are potentially irresolute. *(ii)* P doesn't require an initial aggregator M, only a distance method d. The other three all rely on M. *(iii)* A distance method d used in E and O needs to return the distance between *all* labellings. In contrast, for P and F it is enough that d is defined only between complete labellings.

If we want to apply MO to our problem of aggregating labellings we need to instantiate the two parameters M and d. Let's look at each in turn.

3.1 Initial aggregation methods

An interesting family of resolute aggregation methods has been defined in [1], namely the *interval aggregation methods*. Formally, let Int_n be the set of *intervals* of non-zero length in $\{0,1,\ldots,n\}$, i.e., $Int_n = \{(k,l) \mid k < l,\ k,l \in \{0,1,\ldots,n\}\}$. Let $Y \subseteq Int_n$ be some subset of distinguished intervals in Int_n. Then we define aggregation method F^Y by setting, for each $\mathscr{A}$, $\mathscr{A}$-profile $\mathbf{L}$ and $a \in Args_{\mathscr{A}}$:

$$[F^Y_{\mathscr{A}}(\mathbf{L})](a) = \begin{cases} \mathtt{x} & \text{if } \mathtt{x} \in \{\mathtt{in}, \mathtt{out}\} \text{ and } (|V^{\mathbf{L}}_{a:\neg\mathtt{x}}|, |V^{\mathbf{L}}_{a:\mathtt{x}}|) \in Y \\ \mathtt{undec} & \text{otherwise,} \end{cases}$$

where, for any $\mathtt{x} \in \{\mathtt{in}, \mathtt{out}, \mathtt{undec}\}$, $V^{\mathbf{L}}_{a:\mathtt{x}}$ denotes $\{i \in Ag \mid L_i(a) = \mathtt{x}\}$. A particular member of this family, which we will use in our examples, is the *credulous* aggregation method [4], denoted by *cio*. This is obtained by taking $Y_{cio} = \{(0,l) \in Int_n \mid l > 0\}$. The credulous method *cio* returns a collective label of in (resp. out) to an argument if at least one agent votes for in (resp. out) while none vote for the opposite label out (resp. in). Otherwise it returns undec.

Interval methods may be characterised by a number of postulates such as *Anonymity*, *Unanimity* and *AF-Independence* (the collective label of a is calculated independently of which other arguments might be present or absent from $\mathscr{A}$). However, despite their simplicity, there is *no* interval method that satisfies *Collective Rationality* (we refer the reader to [1] for the details).

3.2 Labelling distance methods

In [2] a few distance methods were proposed, although that paper was concerned only in defining distances between *complete $\mathscr{A}$-labellings* rather than between any two arbitrary $\mathscr{A}$-labellings. (This is a point we will return to later.) The idea is first to define a distance *diff* over the set $\{\mathtt{in}, \mathtt{out}, \mathtt{undec}\}$ of *labels* and then define the distance between two $\mathscr{A}$-labellings as a sum over some set of arguments of the distances between labels assigned by those $\mathscr{A}$-labellings to arguments from the set. Formally, all of the distance methods of [2] shared the following form, given $\mathscr{A}$ and $L_1, L_2 \in Comp(\mathscr{A})$:

$$d^{diff,\mathfrak{S}}_{\mathscr{A}}(L_1, L_2) = \sum_{a \in \mathfrak{S}(\mathscr{A})} diff(L_1(a), L_2(a)) \tag{1}$$

where *(i) diff* is a metric over the set of labels and *(ii)* $\mathfrak{S}$ is a function that, for each AF $\mathscr{A}$ selects a subset $A \subseteq Args_{\mathscr{A}}$ of "important" arguments in $\mathscr{A}$. A number of different combinations of *diff* and $\mathfrak{S}$ were considered in [2].

In particular two distance metrics between labels were considered - *discrete* metrics, i.e. $diff_D(\mathtt{x},\mathtt{y}) = 1$ if $\mathtt{x} \neq \mathtt{y}$, $diff_D(\mathtt{x},\mathtt{y}) = 0$ if $\mathtt{x} = \mathtt{y}$ (which, when plugged into the method sd^{diff} below results in taking the Hamming distance between labellings), and a distance metric which assigns 2 to the hard conflict, i.e. $diff_{rh}(\mathtt{in},\mathtt{out}) = 2$, and 1 to soft conflicts, i.e. $diff_{rh}(\mathtt{in},\mathtt{undec}) = diff_{rh}(\mathtt{out},\mathtt{undec}) = 1$ (which when plugged into sd^{diff} results in the *refined Hamming* distance - see [2]).

Regarding $\mathfrak{S}$, three options were considered. The most obvious idea is to take $\mathfrak{S}(\mathscr{A}) = Args_{\mathscr{A}}$. We denote by sd^{diff} the resulting distance method. As was shown in [2][Sect. 6], the drawback with this method is that it somehow leads to "double-counting" of label differences that are already in some sense implied by others. This leads us to focus on critical sets (originally due to [6] but here generalised to make it relative to $\mathscr{X}$).

Definition 10. *Given $\mathscr{A}$ and $\mathscr{X} \in \{Labs, Comp\}$, a set of arguments $A \subseteq Args_{\mathscr{A}}$ is* $\mathscr{X}$-critical (for $\mathscr{A}$) *iff for any $L_1, L_2 \in \mathscr{X}(\mathscr{A})$, if $L_1[A] = L_2[A]$ then $L_1 = L_2$.*

A $\mathscr{X}$-critical set is a set of arguments such that any two $\mathscr{X}$-labellings are different iff they label at least one argument in the set differently. Note there is only one *Labs*-critical set for $\mathscr{A}$, namely $Args_{\mathscr{A}}$. An idea explored in [2] was for $\mathfrak{S}(\mathscr{A})$ to always select some $\subseteq$*-minimal Comp*-critical set for $\mathscr{A}$. This leads to the definition of the *critical sets distance method*, denoted by $cd^{diff,\mathfrak{S}}$. A problem with this is that different choices of minimal critical set can yield quite different results. The distance method finally arrived at in [2] uses the notion of *issue* (again generalised below to make it relative to $\mathscr{X}$).

Definition 11. *Given $\mathscr{X} \in \{Labs, Comp\}$ we define an equivalence relation $\equiv_{\mathscr{X}}$ over $Args_{\mathscr{A}}$ by setting $a \equiv_{\mathscr{X}} b$ iff* either *$[L(a) = L(b)$ for all $L \in \mathscr{X}(\mathscr{A})]$* or *$[L(a) = \neg L(b)$ for all $L \in \mathscr{X}(\mathscr{A})]$. Each $\equiv_{\mathscr{X}}$-equivalence class is called an* $\mathscr{X}$-issue*, and we denote the $\equiv_{\mathscr{X}}$-equivalence class to which a belongs by $[a]_{\mathscr{X}}$.*

If $a \equiv_{\mathscr{X}} b$ then either a, b are always labelled the same by every labelling in $\mathscr{X}$, or they are always labelled "opposite". Thus changing the label of one of them always produces a change of equal magnitude in the label of the other. The idea behind the *issue-based distance method* $id^{diff,\mathfrak{S}}$ is to form $\mathfrak{S}(\mathscr{A})$ by selecting one representative from each *Comp*-issue. Note $\mathfrak{S}(\mathscr{A})$ so defined will return a *Comp*-critical set for $\mathscr{A}$, though not necessarily a minimal one. It was also shown in [2] that the resulting distance is independent of the choice of representative from the *Comp*-issue. In [2] it was shown that $id^{diff,\mathfrak{S}}$ forms a metric over $Comp(\mathscr{A})$. Below we give a more general result.

Proposition 1. *Let $\mathscr{X} \in \{Labs, Comp\}$. For any label metric diff and any function $\mathfrak{S}$, the function $d^{diff,\mathfrak{S}}_{\mathscr{A}}$ defined in (1) defines a pseudometric over $\mathscr{X}(\mathscr{A})$. Moreover, it defines a metric over $\mathscr{X}(\mathscr{A})$ iff $\mathfrak{S}(\mathscr{A})$ is $\mathscr{X}$-critical.*

In [2] it was assumed the above distances were defined only between complete labellings. In this case $id^{diff,\mathfrak{S}}$ seems to be a good candidate to use in MO. But two of the MO aggregation methods, namely `E` and `O`, require distance to be defined between *all* labellings, and in this case Prop. 1 gives us a problem, for it tells us that the **only** way for any distance method of the form (1) to yield a metric over the whole set $Labs(\mathscr{A})$ is if $\mathfrak{S}(\mathscr{A}) = Args_{\mathscr{A}}$. The question is, is there any alternative way to define a distance method such that $d_{\mathscr{A}}$ is a metric over $Labs(\mathscr{A})$, but which agrees with $id^{diff,\mathfrak{S}}$ on $Comp(\mathscr{A})$? Here we give one possibility. The idea is to take a sum over all arguments, but to weight the contribution of a in the sum by the inverse of the size of the *Comp*-issue to which a belongs. This gives rise to the *extended issue-based distance method* eid^{diff}.

$$eid^{diff}_{\mathscr{A}}(L_1, L_2) = \sum_{a \in Args_{\mathscr{A}}} \frac{diff(L_1(a), L_2(a))}{|[a]_{Comp}|}$$

Proposition 2. (i). *$eid^{diff}_{\mathscr{A}}$ is a metric over $Labs(\mathscr{A})$* (ii). *$eid^{diff}_{\mathscr{A}}(L_1, L_2) = id^{diff,\mathfrak{S}}_{\mathscr{A}}(L_1, L_2)$ for all $L_1, L_2 \in Comp(\mathscr{A})$.*

Therefore we have two distance methods that can be used freely on $Labs(\mathscr{A})$: sd^{diff} and eid^{diff}.

Example 2. *Table 1 presents some distances returned by $d = eid^{diff}_{\mathscr{A}_2}$, where $diff = diff_{rh}$ and $\mathscr{A}_2$ is from Fig. 1. $Args_{\mathscr{A}_2}$ partitions into three Comp-issues $\{a,b\}, \{c\}$ and $\{d,e\}$. Let us calculate a few entries as an example: $d(l_1, l_2) = 2$ because there is no conflict over first two issues and there is a hard conflict over the last one $(0+0+2)$; $d(l_3, l_4) = 2$ because there are two soft conflicts over the first and the last issue and no conflict on the middle one $(1+0+1)$; $d(l_5, n_4) = 1.5$ because there is half of a soft conflict over the first issue and a soft conflict over the second one $(0.5+1+0)$; etc.*

	l_1	l_2	l_3	l_4	l_5	l_6	n_1	n_2	n_3	n_4
l_1 : oioio	0.0	2.0	1.0	1.0	4.0	3.0	2.0	5.0	5.0	4.5
l_2 : oiooi	2.0	0.0	1.0	3.0	2.0	3.0	2.0	3.0	3.0	2.5
l_3 : oiouu	1.0	1.0	0.0	2.0	3.0	2.0	1.0	4.0	4.0	3.5
l_4 : uuoio	1.0	3.0	2.0	0.0	3.0	2.0	1.0	4.0	4.0	4.5
l_5 : uuuoi	4.0	2.0	3.0	3.0	0.0	1.0	2.0	1.0	1.0	1.5
l_6 : uuuuu	3.0	3.0	2.0	2.0	1.0	0.0	1.0	2.0	2.0	2.5
n_1 : uuouu	2.0	2.0	1.0	1.0	2.0	1.0	0.0	3.0	3.0	3.5
n_2 : iouoi	5.0	3.0	4.0	4.0	1.0	2.0	3.0	0.0	2.0	2.5
n_3 : uuioi	5.0	3.0	4.0	4.0	1.0	2.0	3.0	2.0	0.0	0.5
n_4 : uiioi	4.5	2.5	3.5	4.5	1.5	2.5	3.5	2.5	0.5	0.0

Table 1: Distance between labellings by eid^{diff} with refined Hamming distance $diff_{rh}$ over the labels.

	$cio(\mathbf{L})$	P($\mathbf{L}$)	E($\mathbf{L}$)			F($\mathbf{L}$)	O($\mathbf{L}$)	
	n_1	l_5	l_3	l_4	l_6	l_3	l_3	l_6
a	u	u	o	u	u	o	o	u
b	u	u	i	u	u	i	i	u
c	o	u	o	o	u	o	o	u
d	u	o	u	i	u	u	u	u
e	u	i	u	o	u	u	u	u

	$cio(\mathbf{L})$	P($\mathbf{L}$)	E($\mathbf{L}$)	F($\mathbf{L}$)	O($\mathbf{L}$)
	n_1	l_5	l_6	l_3	l_6
a	u	u	u	o	u
b	u	u	u	i	u
c	o	u	u	o	u
d	u	o	u	u	u
e	u	i	u	u	u

Table 2: Aggregation of the profile $\mathbf{L} = (l_4 : \texttt{uuoio}, l_5 : \texttt{uuuoi}, l_5 : \texttt{uuuoi})$ - outcomes for different MO aggregation methods used with: extended issue-based labelling distance (left table), and sum over all arguments (right table). In both columns refined hamming distance over the labels and credulous initial operator were used.

3.3 Example of the MO methods

We illustrate the MO methods in our setting by continuing with the AF $\mathscr{A}_2$ from Fig. 1. The $\mathscr{A}_2$-profile $\mathbf{L} = (l_4 : \texttt{uuoio}, l_5 : \texttt{uuuoi}, l_5 : \texttt{uuuoi})$ aggregated with *cio* results in the non-complete labelling $n_1 : \texttt{uuouu}$. The result of repairing it with MO methods is listed in Table 2.

In the left column the MO methods were instantiated with eid^{diff} with $diff = diff_{rh}$. Method P returns the closest complete labelling to the profile $\mathbf{L}$. We calculate the distance between $\mathbf{L}$ and labellings $l_1 - l_6$ by adding distances from the row l_4 (Table 1) to the doubled distances from the row l_5 and receive $9,7,8,6,3,4$ respectively. The minimum is obtained for labelling $l_5 : \texttt{uuuoi}$.

The E procedure returns the closest complete labelling to $cio(\mathbf{L})$. We inspect the row n_1 in Table 1 to find that the minimum distance 1 is obtained for l_3, l_4 and l_6.

The F and O procedures search for closest *cio*-consistent profiles $\mathbf{L}'$ to profile $\mathbf{L}$. The F procedure is restricted to the complete *cio*-consistent profiles. Consider $\mathbf{L}' = (l_1 : \texttt{oioio}, l_5 : \texttt{uuoio}, l_5 : \texttt{uuoio})$. It is a *cio*-consistent profile with $cio(\mathbf{L}') = l_3 : \texttt{oiouu}$. It is also minimal. The profiles $\mathbf{L}$ and $\mathbf{L}'$ differ just on l_1 and l_4 with distance 1 (soft conflict on issue $\{a,b\}$). There are no other complete *cio*-consistent profiles with different *cio* outcome and same distance, because any change of labelling l_4 to another complete labelling costs more than 1. The only other candidate for change is labelling l_5. It can be changed to $l_6 : \texttt{uuuuu}$ for a cost of 1 but to affect the outcome of *cio* both occurrences of l_5 in $\mathbf{L}$ would have to be changed with a total cost of 2. The profile $\mathbf{L}'$ works for the O procedure as well but O is not restricted to complete *cio*-consistent profiles. Changing one of the l_5 labellings to a non-complete labelling $n_3 : \texttt{uuioi}$ with distance 1 creates additional conflict on argument c. As a result the labelling $l_6 : \texttt{uuuuu}$ is produced.

The results change when we switch to using sd^{diff} (with $diff = diff_{rh}$) rather than eid^{diff} (right column). In this case size of the issues does matter. The E procedure only selects l_6 because it differs with n_1 over issue $\{c\}$ with one argument, while the other two labellings l_3, l_4 differ over the two-argument issues $\{a,b\}$ and $\{d,e\}$ respectively. Similarly in the case of O changing l_5 into n_3 over a single-argument issue is closer than the change of l_4 into l_1 over an issue with two arguments.

4 Qualitative distance postulates

In this section we consider two new postulates for distance measures in argumentation, which are inspired by similar postulates considered in the binary aggregation of MO.

One intuition that MO have about distance between judgment sets in their setting is that the *quantitative* distance between two judgment sets X,Y, expressed by $d(X,Y)$, should depend only on the *qualitative distance* between them. In their binary setting in which every judgment set either accepts or rejects each agenda item, the natural way to measure such qualitative distance is simply to take the set of agenda items that X and Y differ on. In our 3-valued setting we can have two labellings that disagree on an argument to differing *degrees*. We thus adapt the notion of qualitative distance as follows:

Definition 12. *Let $\mathscr{A}$ be an AF and $L_1,L_2 \in Labs(\mathscr{A})$. The* qualitative distance between L_1 and L_2, *denoted $L_1 \ominus L_2$, is defined as $L_1 \ominus L_2 = \langle C(L_1,L_2), H(L_1,L_2)\rangle$, where $C(L_1,L_2)$ and $H(L_1,L_2)$ are resp. the sets of* conflicts *and of* hard conflicts *between L_1 and L_2, i.e., $C(L_1,L_2) \stackrel{\text{def}}{=} \{a \in Args_{\mathscr{A}} \mid L_1(a) \neq L_2(a)\}$ and $H(L_1,L_2) \stackrel{\text{def}}{=} \{a \in Args_{\mathscr{A}} \mid L_1(a) = \neg L_2(a) \neq \texttt{undec}\}$.*

Note that $H(L_1,L_2) \subseteq C(L_1,L_2)$. We can define a natural ordering between different qualitative distances as follows:

$$L_1 \ominus L_2 \preceq N_1 \ominus N_2 \text{ iff } [C(L_1,L_2) \subseteq C(N_1,N_2) \text{ and } H(L_1,L_2) \subseteq H(N_1,N_2)]$$

So if every conflict, resp. hard conflict, between L_1,L_2 remains a conflict, resp. hard conflict, between N_1,N_2 then the disagreement between N_1,N_2 is qualitatively at least as large as that between L_1,L_2. One can check that $\preceq$ forms a preorder on $Labs(\mathscr{A})^2$. We denote the strict part of $\preceq$ by $\prec$. We can now adapt the distance property of *Normality* [9] to our setting. It expresses that a (weak) increase in qualitative distance should yield a (weak) increase in quantitative distance.

Normality If $L_1 \ominus L_2 \preceq N_1 \ominus N_2$ then $d_{\mathscr{A}}(L_1,L_2) \leq d_{\mathscr{A}}(N_1,N_2)$

If *Normality* holds then clearly we also have that if $L_1 \ominus L_2 = N_1 \ominus N_2$ then $d_{\mathscr{A}}(L_1,L_2) = d_{\mathscr{A}}(N_1,N_2)$. Thus if *Normality* is required then the problem of defining a distance measure $d_{\mathscr{A}}$ for $\mathscr{A}$ essentially reduces to the problem of assigning a number to $L_1 \ominus L_2$ for each $L_1,L_2 \in Labs(\mathscr{A})^2$.

Example 3. *For $\mathscr{A}_2$ from Fig. 1 we have $l_2 \ominus l_3 = l_5 \ominus l_6 = \langle \{d,e\}, \emptyset\rangle \preceq \langle \{d,e\}, \{d,e\}\rangle = l_1 \ominus l_2$. Hence if d satisfies* Normality *we must have $d_{\mathscr{A}_2}(l_2,l_3) = d_{\mathscr{A}_2}(l_5,l_6) \leq d_{\mathscr{A}_2}(l_1,l_2)$.*

Similarly we can postulate that a *strict* increase in qualitative distance should lead to a *strict* increase in quantitative distance, leading to the following adaptation of the *Increasing* property of MO:

Increasing If $L_1 \ominus L_2 \prec N_1 \ominus N_2$ then $d_{\mathscr{A}}(L_1,L_2) < d_{\mathscr{A}}(N_1,N_2)$

How do the distance methods described in the previous section fare with respect to *Normality* and *Increasing*? As we see, provided *diff* satisfies certain conditions then at least some of them validate these postulates.

Proposition 3. (i). *Assume the label metric diff satisfies $diff(\texttt{in},\texttt{out}) \geq diff(\texttt{in},\texttt{undec}) = diff(\texttt{out},\texttt{undec})$. Then all of $sd^{diff}_{\mathscr{A}}$, $cd^{diff,\mathfrak{S}}_{\mathscr{A}}$, $id^{diff,\mathfrak{S}}_{\mathscr{A}}$ and $eid^{diff}_{\mathscr{A}}$ satisfy* Normality *over $Labs(\mathscr{A})$ (and hence also over $Comp(\mathscr{A})$).*
(ii). *Assume the label metric diff satisfies $diff(\texttt{in},\texttt{out}) > diff(\texttt{in},\texttt{undec}) = diff(\texttt{out},\texttt{undec})$. Then* (1). *$sd^{diff}_{\mathscr{A}}$ and $eid^{diff}_{\mathscr{A}}$ satisfy* Increasing *over $Labs(\mathscr{A})$ (and hence also $Comp(\mathscr{A})$).* (2). *$id^{diff,\mathfrak{S}}_{\mathscr{A}}$ satisfies* Increasing *over $Comp(\mathscr{A})$, but not over $Labs(\mathscr{A})$.* (3) *$cd^{diff,\mathfrak{S}}_{\mathscr{A}}$ does not satisfy* Increasing *over $Comp(\mathscr{A})$ (and hence also not over $Labs(\mathscr{A})$).*

Observe that $diff_{rh}$ satisfies both conditions mentioned in *(i)*, *(ii)* above, and so can be plugged into sd^{diff} or eid^{diff} to obtain a distance method satisfying both *Normality* and *Increasing*.

Surprisingly, $cd^{diff,\mathfrak{S}}$ fails *Increasing* even when restricted to $Comp(\mathscr{A})$, as the next example demonstrates.

Example 4. *Consider $\mathscr{A}_1$ from Fig. 1. There are three Comp-issues: $\{a,b\},\{c\},\{d,e\}$. But the label of c is determined by the labels of the other arguments, so there are 4 possible minimal Comp-critical sets $\{a,b\} \times \{d,e\}$. Let's take $\{a,d\}$ as our selected minimal set (though the counterexample will also work for any of the other three possible choices). Consider labellings l_1: oioio, l_8: ioioi and l_2: oiooi, l_7: iooio. We have $l_2 \ominus l_7 = \langle \{a,b,d,e\}, \{a,b,d,e\}\rangle \prec \langle \{a,b,c,d,e\}, \{a,b,c,d,e\}\rangle = l_1 \ominus l_8$ but $d(l_2,l_7) = d(l_1,l_8) = 2 \times diff(\texttt{in},\texttt{out})$.*

5 Conclusions

We have continued work on distance methods for argumentation that was initiated in [2], illustrating how they can be employed to address problems of aggregation in argumentation. To do this we adapted the framework of Miller and Osherson from binary judgment aggregation to our setting, defining several operators for aggregating argument labellings. In the process we extended and generalised some results on the distance measures from [2]. We also examined an adaptation of the MO postulates *Normality* and *Increasing* to our setting and were able to identify the circumstances under which they hold. Under some mild conditions on the label metric *diff*, it turned out that only two of our distance methods, namely sd^{diff} and the *extended issue-based* method eid^{diff} are able to satisfy both without restriction on their domain $Labs(\mathscr{A})$.

There are several avenues open for future work. Firstly, a feature of our examples (see Sect. 3.3) is that the different aggregation methods can all yield quite different results. This raises the question of which method to prefer. We plan to classify the different methods in terms of the postulates they satisfy. Previous works on labelling aggregation [1, 10] have examined postulates for such operators (inspired by postulates from JA), but have done so only for *resolute* aggregation methods. We will generalise these to irresolute methods, perhaps taking a lead from similar generalisations from JA [7].

Secondly, it can be shown that aggregation methods which use the *down-admissible* and *up-complete* procedures of [4] can be represented as an instance of Endpoint method. We would like to investigate the classes of aggregation operators given by different distance methods and relate them to the ones which use *down-admissible* and *up-complete* procedures.

Another interesting question is to consider what happens if you aggregate *all* complete $\mathscr{A}$-labellings of an AF. This question was considered in [4], but again only for resolute operators. In this way the operators of [4] were able to characterise certain *single-status* argumentation semantics (i.e., grounded and ideal). Our move to irresolute aggregation opens the possibility that we might be able to capture also some *multiple-status* semantics such as *preferred* [5]. That is, does aggregating all complete $\mathscr{A}$-labellings yield precisely the set of preferred $\mathscr{A}$-labellings?

References

[1] R. Booth, E. Awad, and I. Rahwan. Interval methods for judgment aggregation in abstract argumentation. In *Proc. KR 2014*, pages 594–597, 2014.

[2] R. Booth, M. Caminada, M. Podlaszewski, and I. Rahwan. Quantifying disagreement in argument-based reasoning. In *Proc. AAMAS 2012*, pages 493–500, 2012.

[3] M. Caminada and D. Gabbay. A logical account of formal argumentation. *Studia Logica*, 93(2-3):109–145, 2009.

[4] M. Caminada and G. Pigozzi. On judgment aggregation in abstract argumentation. *Autonomous Agents and Multi-Agent Systems*, 22(1):64–102, 2011.

[5] P.M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77(2):321–357, 1995.

[6] D. Gabbay. Fibring argumentation frames. *Studia Logica*, 93(2-3):231–295, 2009.

[7] U. Grandi and G. Pigozzi. On compatible multi-issue group decisions. *Proc. LOFT-2012*, 2012.

[8] C. List and C. Puppe. Judgment aggregation: A survey. In *Handbook of Rational and Social Choice*. Oxford University Press, 2009.

[9] M. Miller and D. Osherson. Methods for distance-based judgment aggregation. *Social Choice and Welfare*, 32(4):575–601, 2009.

[10] I. Rahwan and F. Tohmé. Collective argument evaluation as judgement aggregation. In *Proc. AAMAS 2010*, pages 417–424, 2010.

Grounded Semantics and Infinitary Argumentation Frameworks

Martin Caminada [a] Nir Oren [a]

[a] *University of Aberdeen, Department of Computing Science*

Abstract

Computing the grounded extension of an argumentation framework can be done using the well-known inductive procedure of Dung's landmark paper. However, this procedure has only been proven to be correct for finitary argumentation frameworks, that is, frameworks in which every argument has only a finite number of defeaters. The problem is that formalisms like ASPIC$^+$ and ASPIC$^-$ can easily generate frameworks in which arguments have an infinite number of defeaters. In the current paper, we will therefore broaden the applicability of the proof procedures for grounded semantics, and weaken the condition that the argumentation framework has to be finitary.

1 Introduction

Rule-based instantiated argumentation formalisms, such as ASPIC [1], ASPIC$^+$ [9, 10] and the argument-interpretation of ABA [6] have enjoyed an increasing popularity within the formal argumentation community. Their main advantage over abstract argumentation (c.f. [5]) is that they enable nonmonotonic entailment to be defined as rule-based inference. This can have advantages when it comes to the ability to explain formal nonmonotonic inference in terms that human actors can relate to, as observed in [2].

One particular difficulty that several rule-based argumentation formalisms are subject to is that even a finite set of rules can lead to an infinite set of arguments. For instance, in an ASPIC type framework [1, 3, 9, 10], and adopting ASPIC notation, given a set of strict rules $\{\rightarrow a; \quad \rightarrow b; \quad \neg c \rightarrow \neg d; \quad \neg d \rightarrow \neg c\}$ and a set of defeasible rules $\{a \Rightarrow c; \quad b \Rightarrow \neg c\}$ the argument $A : (\rightarrow a) \Rightarrow c$ has an infinite number of defeaters: $B_1 : (\rightarrow b) \Rightarrow \neg c$, $B_2 : (((\rightarrow b) \Rightarrow \neg c) \rightarrow \neg d) \rightarrow \neg c$, $B_3 : (((((\rightarrow b) \Rightarrow \neg c) \rightarrow \neg d) \rightarrow \neg c) \rightarrow \neg d) \rightarrow \neg c$, etc. Similar observations can be made for the argumentation interpretation of ABA [6] and the argumentation interpretation of logic programming [12].

The above example illustrates that even a finite knowledge base can lead to an argumentation framework that is not only infinite, but that is even infinitary in the sense of [5].[1] This can be problematic, as some of the fundamental results of abstract argumentation have only been proven for finitary argumentation frameworks, for example the existence of semi-stable and stage extensions [11] and the correctness of the inductive procedure for computing the grounded extension [5].

One particular way in which this problem has been dealt with is by adding an extra constraint to the argument construction process, so that each rule can be used at most once within each "branch" of an argument.[2] Hence, in the above example B_3 would no longer form a well-formed argument. The disadvantage of this approach, however, is that argument construction loses some of its modular aspects. For instance, if one has an argument A with conclusion a, an argument B with conclusion b, and a rule $a, b \rightarrow c$ then one can no longer be sure that $A, B \rightarrow c$ is a well-formed argument. This can cause difficulties for work like [3] where part of the technical results relies on modular argument construction.

[1] Recall that an argumentation framework is finite when it has a finite number of arguments. It is finitary when each argument has a finite number of defeaters.

[2] Recall that the recursive definition of an argument in ASPIC, ASPIC$^+$ and ASPIC$^-$ essentially defines a tree of rules, similar to what is done in the argumentation interpretation of ABA.

Ideally, one would like to have a solution that does not in any way restrict the construction of arguments. However, this requires the broadening of some of the fundamental results of abstract argumentation theory to particular classes of infinitary argumentation frameworks. In the current paper, we introduce such a broadening. In particular, we show that for argumentation frameworks generated by the ASPIC$^-$ formalism (which, as we have seen, can be infinitary) the iterative procedure for computing the grounded extension is correct as far as the conclusions are concerned. Hence, when it comes to determining the outcome of ASPIC$^-$ under grounded semantics (in terms of the conclusions yielded) one is free to apply the inductive definition of grounded semantics, even though there may be differences on the argument level.

The remainder of this paper is structured as follows. First, in Section 2, we introduce the formal preliminaries of abstract argumentation. Then, in Section 3, we study the effects of omitting what we call "superseded" arguments. In Section 4 we then use our results to show that under ASPIC$^-$ the inductive definition of grounded semantics yields the same conclusions as the grounded extension itself, even though the underlying argumentation framework may not be finitary. We then round off with a discussion of the obtained results in Section 5.

2 Formal Preliminaries

In the current section, we briefly restate some of the key concepts of abstract argumentation theory.

Definition 1 ([5])**.** *An* argumentation framework *is a pair* (Ar, def) *where* Ar *is set of entities, called arguments, whose internal structure can be left unspecified, and* def *a binary relation on* Ar*. We say that* A defeats B *iff* $(A, B) \in def$*. We say that the argumentation framework is* finite *iff* Ar *is finite. We say that the argumentation framework is* finitary *iff for every* $A \in Ar$*,* $\{B \in Ar \mid (B, A) \in def\}$ *is finite.*

Definition 2. *Let* $AF = (Ar, def)$ *be an argumentation framework,* $A \in Ar$ *and* $\mathcal{A}rgs \subseteq Ar$*. We define* A^+ *as* $\{B \in Ar \mid A$ *defeats* $B\}$*,* A^- *as* $\{B \in Ar \mid B$ *defeats* $A\}$*,* $\mathcal{A}rgs^+$ *as* $\bigcup\{A^+ \mid A \in \mathcal{A}rgs\}$*, and* $\mathcal{A}rgs^-$ *as* $\bigcup\{A^- \mid A \in \mathcal{A}rgs\}$*.* $\mathcal{A}rgs$ *is said to be* conflict-free *iff* $\mathcal{A}rgs \cap \mathcal{A}rgs^+ = \emptyset$*.* $\mathcal{A}rgs$ *is said to* defend A *iff* $A^- \subseteq \mathcal{A}rgs^+$*. The characteristic function* $F_{AF} : 2^{Ar} \to 2^{Ar}$ *is defined as* $F_{AF}(\mathcal{A}rgs) = \{A \in Ar \mid \mathcal{A}rgs$ *defends* $A\}$*.*

Definition 3. *Let* $AF = (Ar, def)$ *be an argumentation framework.* $\mathcal{A}rgs \subseteq Ar$ *is said to be:*

- *an admissible set iff* $\mathcal{A}rgs$ *is conflict-free and* $\mathcal{A}rgs \subseteq F_{AF}(\mathcal{A}rgs)$
- *a complete extension iff* $\mathcal{A}rgs$ *is conflict-free and* $\mathcal{A}rgs = F_{AF}(\mathcal{A}rgs)$
- *a grounded extension iff* $\mathcal{A}rgs$ *is the smallest (w.r.t.* $\subseteq$*) complete extension*

3 Omitting Superseded Arguments

The idea of superseded arguments is to identify those arguments that can be omitted from the argumentation framework without significantly affecting its outcome, as long as for each argument one omits, one keeps an argument that supersedes it.

Definition 4 (argument superseding)**.** *An argument* A *is* superseded *by an argument* B *iff* $A^+ \subseteq B^+$ *and* $A^- \supseteq B^-$*.*

Please notice that the supersedes relationship among arguments is not a partial order because it does not satisfy anti-symmetry. Hence, it does *not* satisfy Postulate 3.1 of [7], so we cannot apply their theory. We now proceed to define the supersedes relationship between argumentation frameworks.

Definition 5 (AF superseding)**.** *Let* $AF = (Ar, def)$ *be an argumentation framework, and let* $Ar' \subseteq Ar$ *be such that for each* $A \in Ar$ *there exists an* $A' \in Ar'$ *that supersedes it. Let* AF' *be* (Ar', def') *with* $def' = def \cap (Ar' \times Ar')$*. We say that* AF' *supersedes* AF*.*

Notice that the supersedes relationship among argumentation frameworks *does* constitute a partial order.

Proposition 1. *Let* $AF = (Ar, def)$ *and* $AF' = (Ar', def')$ *be argumentation frameworks such that* AF' *supersedes* AF*, and let* $\mathcal{A}rgs' \subseteq Ar'$*. It holds that* $F_{AF'}(\mathcal{A}rgs') \subseteq F_{AF}(\mathcal{A}rgs')$*.*

Proof. Let $A \in F_{AF'}(\mathcal{A}rgs')$. So each $B' \in Ar'$ that defeats A is defeated by some $C \in \mathcal{A}rgs'$. Let $B \in Ar$ be an argument that defeats A. Let $B' \in Ar'$ be an argument that supersedes B. Then, from the fact that $B^+ \subseteq B'^+$ it follows that B' also defeats A. Hence, B' is defeated by some $C \in \mathcal{A}rgs'$. Since $B^- \supseteq B'^-$ it follows that this C also defeats B. Hence, A is defended by $\mathcal{A}rgs'$ under AF. That is, $A \in F_{AF}(\mathcal{A}rgs')$. □

Proposition 2. *Let $AF = (Ar, def)$ and $AF' = (Ar', def')$ be argumentation frameworks such that AF' supersedes AF, and let $\mathcal{A}rgs' \subseteq Ar'$. It holds that $F_{AF}(\mathcal{A}rgs') \cap Ar' = F_{AF'}(\mathcal{A}rgs')$.*

Proof.

$F_{AF'}(\mathcal{A}rgs') \subseteq F_{AF}(\mathcal{A}rgs') \cap Ar'$
: Proposition 1 states that $F_{AF'}(\mathcal{A}rgs') \subseteq F_{AF}(\mathcal{A}rgs')$, so from $F_{AF'}(\mathcal{A}rgs') \subseteq Ar'$ it then follows that $F_{AF'}(\mathcal{A}rgs') \subseteq F_{AF}(\mathcal{A}rgs') \cap Ar'$.

$F_{AF}(\mathcal{A}rgs') \cap Ar' \subseteq F_{AF'}(\mathcal{A}rgs')$
: Let $A \in F_{AF}(\mathcal{A}rgs') \cap Ar'$. The fact that $A \in F_{AF}(\mathcal{A}rgs')$ means that each $B \in Ar$ that defeats A is defeated by some $C \in \mathcal{A}rgs'$. The fact that $Ar' \subseteq Ar$ implies that also each $B' \in Ar'$ that defeats A is defeated by some $C \in \mathcal{A}rgs'$. Hence, $\mathcal{A}rgs'$ defends $A \in Ar'$ under AF'. That is, $A \in F_{AF'}(\mathcal{A}rgs')$. □

The complete extensions of a superseded argumentation framework can be converted to the extensions of the superseding argumentation framework, and vice versa.

Theorem 1. *Let $AF = (Ar, def)$ and $AF' = (Ar', def')$ be argumentation frameworks such that AF' supersedes AF.*

1. *if CE is a complete extension of AF, then $CE \cap Ar'$ is a complete extension of AF'*
2. *if CE' is a complete extension of AF', then $F_{AF}(CE')$ is a complete extension of AF*
3. *if CE is a complete extension of AF, then $F_{AF}(CE \cap Ar') = CE$*
4. *if CE' is a complete extension of AF', then $F_{AF}(CE') \cap Ar' = CE'$*

Proof.

1. Let CE be a complete extension of AF and let CE' be $CE \cap Ar'$. We need to prove that CE' is a conflict-free fixed-point of $F_{AF'}$. Conflict-freeness follows from the fact that CE is conflict-free and $CE' \subseteq CE$. To prove that CE' is a fixed-point of $F_{AF'}$ we need to show two things:

   $CE' \subseteq F_{AF'}(CE')$
   : Let $A \in CE'$. Then the facts that $A \in CE$ and CE is a complete extension imply that each $B \in Ar$ that defeats A is defeated by some $C \in CE$. From $Ar' \subseteq Ar$ it then follows that each $B' \in Ar'$ that defeats A is defeated by some $C \in CE$. The fact that AF' supersedes AF implies that there is a $C' \in Ar'$ with $C^+ \subseteq C'^+$, so C' defeats B'. Since this C' is defended by CE (since the facts that CE is a complete extension and $C \in CE$ imply that C is defended by CE, so the fact that $C^- \supseteq C'^-$ implies that C' is also defended by CE) it follows that $C' \in CE$, so $C' \in CE \cap Ar'$. That is, $C' \in CE'$, so $A \in F_{AF'}(CE')$.

   $F_{AF'}(CE') \subseteq CE'$
   : Let $A \in F_{AF'}(CE')$. From $CE' \subseteq CE$ it follows that $F_{AF}(CE') \subseteq F_{AF}(CE)$ (since F_{AF} is a monotonic function). As $F_{AF'}(CE') \subseteq F_{AF}(CE')$ (Proposition 1) it follows (by transitivity of $\subseteq$) that $F_{AF'}(CE') \subseteq F_{AF}(CE)$. As CE is a complete extension of AF, it holds that $F_{AF}(CE) = CE$, so we obtain $F_{AF'}(CE') \subseteq CE$. Since, by definition, $F_{AF'}(CE') \subseteq Ar'$ it then follows that $F_{AF'}(CE') \subseteq CE \cap Ar'$. That is, $F_{AF'}(CE') \subseteq CE'$.

2. Let CE' be a complete extension of AF'. We need to prove that $F_{AF}(CE')$ is a conflict-free fixed-point of F_{AF}. We first show that $F_{AF}(CE')$ is conflict-free. Suppose, towards a contradiction, that $F_{AF}(CE')$ is not conflict-free. That is, there exist $A, B \in F_{AF}(CE')$ such that A defeats B. Then CE' contains an argument C that defeats A (this is because CE' defends B).

However, the fact that CE' also defends A implies that CE' also contains an argument D that defeats C. But then CE' is not conflict-free, so CE' is not a complete extension of AF'. Contradiction.
We proceed to show that $F_{AF}(CE')$ is a fixed-point of F_{AF}. That is, $F_{AF}(CE') = F_{AF}(F_{AF}(CE'))$.

$F_{AF}(CE') \subseteq F_{AF}(F_{AF}(CE'))$
From the fact that CE' is a complete extension of AF' it follows that $CE' \subseteq F_{AF'}(CE)$. Since $F_{AF'}(CE') \subseteq F_{AF}(CE')$ (Proposition 1) it then follows (transitivity $\subseteq$) that $CE' \subseteq F_{AF}(CE')$. From the fact that F_{AF} is a monotonic function it then follows that $F_{AF}(CE') \subseteq F_{AF}(F_{AF}(CE'))$.

$F_{AF}(F_{AF}(CE')) \subseteq F_{AF}(CE')$
Let $A \in F_{AF}(F_{AF}(CE'))$. Then each $B \in Ar$ that defeats A is defeated by some $C \in F_{AF}(CE')$. Let $C' \in Ar'$ be an argument that supersedes C. From the facts that C is defended by CE' and $C^- \supseteq C'^-$ it follows that C' is also defended by CE'. That is, $C' \in F_{AF}(CE')$. Since $C' \in Ar'$ it then follows that $C' \in F_{AF}(CE') \cap Ar'$. So A is defended by $F_{AF}(CE') \cap Ar'$. That is, $A \in F_{AF}(F_{AF}(CE') \cap Ar')$. Proposition 2 states that $F_{AF}(CE') \cap Ar' = F_{AF'}(CE')$ so we obtain that $A \in F_{AF}(F_{AF'}(CE'))$. But since CE' is a complete extension of AF' it holds that $F_{AF'}(CE') = CE'$, so $A \in F_{AF}(CE')$.

3. Let CE be a complete extension of AF. We need to prove that $CE = F_{AF}(CE \cap Ar')$

$CE \subseteq F_{AF}(CE \cap Ar')$
Let $A \in CE$. Then, from the fact that CE is a complete extension of AF, it follows that for each $B \in Ar$ that defeats A, there is a $C \in CE$ that defeats B. Let $C' \in Ar'$ be an argument that supersedes C. From the fact that $C^+ \subseteq C'^+$ it follows that C' defeats B. The fact that $C \in CE$ means that C is defended by CE (as CE is a complete extension) so from the fact that $C^- \supseteq C'^-$ it follows that C' is also defended by CE. Hence, $C' \in CE$, so $C' \in CE \cap Ar'$. So A is defended by $CE \cap Ar'$. That is, $A \in F_{AF}(CE \cap Ar')$.

$F_{AF}(CE \cap Ar') \subseteq CE$
It trivially holds that $CE \cap Ar' \subseteq CE$. Since F_{AF} is a monotonic function, it then follows that $F_{AF}(CE \cap Ar') \subseteq F_{AF}(CE)$. Since CE is a complete extension of AF, it holds that $F_{AF}(CE) = CE$. Hence, $F_{AF}(CE \cap Ar') \subseteq CE$.

4. Let CE' be a complete extension of AF'. We need to prove that $F_{AF}(CE') \cap Ar' = CE'$.

$CE' \subseteq F_{AF}(CE') \cap Ar'$
Let $A \in CE'$. Then, by definition, $A \in Ar'$. The fact that CE' is a complete extension of AF' means that A is defended by CE' (under AF'). So each $B' \in Ar'$ that defeats A is defeated by some $C \in CE'$. We now show that each $B \in Ar$ that defeats A is defeated by some $C \in CE'$. Let $B \in Ar$ be an argument that defeats A. Let $B' \in Ar'$ be an argument that supersedes B. From the fact that $B^+ \subseteq B'^+$ it follows that B' defeats A. So there exists a $C \in CE'$ that defeats B'. Since $B^- \supseteq B'^-$ it follows that C defeats B. So A is defended (under AF) by CE'. That is, $A \in F_{AF}(CE')$. This, together with the earlier observed fact that $A \in Ar'$ implies that $A \in F_{AF}(CE') \cap Ar'$.

$F_{AF}(CE') \cap Ar' \subseteq CE'$
Let $A \in F_{AF}(CE') \cap Ar'$. Then, the fact that $A \in F_{AF}(CE')$ implies that each $B \in Ar$ that defeats A is defeated by some $C \in CE'$. From the fact that $Ar' \subseteq Ar$ it follows that also each $B' \in Ar'$ that defeats A is defeated by some $C \in CE'$, so CE' defends A under AF'. That is, $A \in F_{AF'}(CE')$. But since CE' is a complete extension of AF', it holds that $F_{AF'}(CE') = CE'$. Hence, $A \in CE'$. □

The grounded extension of a superseded argumentation framework can be converted to the grounded extension of the superseding argumentation framework, and vice versa.

Theorem 2. *Let $AF = (Ar, def)$ and $AF' = (Ar', def')$ be argumentation frameworks such that AF' supersedes AF.*

1. *If GE is the grounded extension of AF, then $GE \cap Ar'$ is the grounded extension of AF'.*

2. *If GE' is the grounded extension of AF', then $F_{AF}(GE')$ is the grounded extension of AF.*

Proof.

1. Let GE be the grounded extension of AF and let GE' be $GE \cap Ar'$. From the fact that GE is also a complete extension of AF, it follows (Theorem 1, point 1) that GE' is a complete extension of AF'. In order to prove that GE' is also the grounded extension of AF', we show that for each complete extension CE' of AF', it holds that $GE' \subseteq CE'$. Let CE' be a complete extension of AF'. Then from Theorem 1 (point 2) it follows that $F_{AF}(CE')$ is a complete extension of AF, so $GE \subseteq F_{AF}(CE')$, which implies that $GE \cap Ar' \subseteq F_{AF}(CE') \cap Ar'$. Theorem 1 (point 4) states that $F_{AF}(CE') \cap Ar' = CE'$, so we obtain that $GE \cap Ar' \subseteq CE'$, so (as $GE' = GE \cap Ar'$) $GE' \subseteq CE'$.

2. Let GE' be the grounded extension of AF', and let GE be $F_{AF}(GE')$. From the fact that GE' is also a complete extension of AF', it follows that GE is a complete extension of AF (Theorem 1, point 2). In order to prove that GE is also the grounded extension of AF, we show that for each complete extension CE of AF, it holds that $GE \subseteq CE$. Let CE be a complete extension of AF. Then (Theorem 1, point 1) $CE \cap Ar'$ is a complete extension of AF'. From the fact that GE' is the grounded extension of AF', it then follows that $GE' \subseteq CE \cap Ar'$. Since F_{AF} is a monotonic function, we obtain $F_{AF}(GE') \subseteq F_{AF}(CE \cap Ar')$. Since $F_{AF}(GE') = GE$ (by definition) and $F_{AF}(CE \cap Ar') = CE$ (Theorem 1, point 3) we obtain that $GE \subseteq CE$. □

4 Omitting C-Superseded Arguments

So far, we have proved equivalence purely on the semantic level (for complete and grounded semantics). The next step is to examine things at the level of proof procedures. Our aim is to examine to what extent one can still apply the iterative procedure for determining grounded semantics in the presence of a possibly infinite argumentation framework that is superseded by a finite argumentation framework. We start with a lemma.

Lemma 1. *Let $AF = (Ar, def)$ and $AF' = (Ar', def')$ be argumentation frameworks such that AF' supersedes AF. For every $i \in \{0, 1, 2, \ldots\}$ it holds that $F^i_{AF'}(\emptyset) \subseteq F^i_{AF}(\emptyset)$.*

Proof. By induction over i:

basis $i = 0$. In that case $F^i_{AF'}(\emptyset) \subseteq F^i_{AF}(\emptyset)$, as $F^0_{AF'}(\emptyset) = \emptyset = F^0_{AF}(\emptyset)$.

step Suppose that $F^i_{AF'}(\emptyset) \subseteq F^i_{AF}(\emptyset)$ for some $i \in \{0, 1, 2, \ldots\}$. As F_{AF} is a monotonic function, it follows that $F_{AF}(F^i_{AF'}(\emptyset)) \subseteq F_{AF}(F^i_{AF}(\emptyset))$. As $F_{AF'}(F^i_{AF'}(\emptyset)) \subseteq F_{AF}(F^i_{AF'}(\emptyset))$ (Proposition 1) we obtain that $F_{AF'}(F^i_{AF'}(\emptyset)) \subseteq F_{AF}(F^i_{AF}(\emptyset))$. That is, $F^{i+1}_{AF'}(\emptyset) \subseteq F^{i+1}_{AF}(\emptyset)$. □

In the context of this work, we are interested in equivalence at the level of conclusions rather than equivalence purely at the level of arguments. For this, we need the following two definitions. Note that if A is an argument we write $\texttt{Conc}(A)$ for its conclusion, and if $\mathcal{A}rgs$ is a set of arguments we write $\texttt{Concs}(\mathcal{A}rgs)$ for $\{\texttt{Conc}(A) \mid A \in \mathcal{A}rgs\}$ as is done in ASPIC$^-$ [3].

Definition 6. *An argument A is* c-superseded *by an argument B iff A is superseded by B and $\texttt{Conc}(A) = \texttt{Conc}(B)$.*

Definition 7. *Let $AF = (Ar, def)$ be an argumentation framework, and let $Ar' \subseteq Ar$ be such that for each $A \in Ar$ there exists an $A' \in Ar'$ that c-supersedes it. Let AF' be (Ar', def') with $def' = def \cap (Ar' \times Ar')$. We say that AF' c-supersedes AF.*

Trivially, it holds that if A is c-superseded by B then A is superseded by B (but not vice versa) and that if AF is c-superseded by AF' then AF is superseded by AF' (but not vice versa). We now come to one of the main results of this paper.

Theorem 3. *Let $AF = (Ar, def)$ be an argumentation framework for which there exists a finitary argumentation framework $AF' = (Ar', def')$ that c-supersedes it. Let GE be the grounded extension of AF. It holds that $\texttt{Concs}(GE) = \texttt{Concs}(\cup_{i=0}^{\infty} F^i_{AF}(\emptyset))$.*

Proof. Let GE' be the grounded extension of AF'. We need to show two things:

$\texttt{Concs}(GE) \subseteq \texttt{Concs}(\cup_{i=0}^{\infty} F_{AF}^{i}(\emptyset))$
Let $a \in \texttt{Concs}(GE)$. Then there is an $A \in GE$ with $\texttt{Conc}(A) = a$. Let $A' \in Ar'$ be an argument that c-supersedes A. From the fact that A is defended by GE (as GE is a complete extension) and $A^- \supseteq A'^-$ it follows that A' is defended by GE, so $A' \in GE$. That is, $A' \in GE \cap Ar'$, so (Theorem 2, point 1) $A' \in GE'$. Since AF' is finitary, it holds that $GE' = \cup_{i=0}^{\infty} F_{AF'}(\emptyset)$, so $A' \in \cup_{i=0}^{\infty} F_{AF'}^{i}(\emptyset)$. From Lemma 1 it follows that $\cup_{i=0}^{\infty} F_{AF'}^{i}(\emptyset) \subseteq \cup_{i=0}^{\infty} F_{AF}^{i}(\emptyset)$ so $A' \in \cup_{i=0}^{\infty} F_{AF}^{i}(\emptyset)$ so $\texttt{Conc}(A') \in \texttt{Concs}(\cup_{i=0}^{\infty} F_{AF}^{i}(\emptyset))$. Since $\texttt{Conc}(A') = \texttt{Conc}(A)$ (as A' c-supersedes A) it then follows that $a \in \texttt{Concs}(\cup_{i=0}^{\infty} F_{AF}^{i}(\emptyset))$.

$\texttt{Concs}(\cup_{i=0}^{\infty} F_{AF}^{i}(\emptyset)) \subseteq \texttt{Concs}(GE)$
As proven by Dung [5], it holds for any argumentation framework AF (finitary or infinitary) that $\cup_{i=0}^{\infty} F_{AF}^{i}(\emptyset) \subseteq GE$. From the fact that Concs is a monotonic function, it then directly follows that $\texttt{Concs}(\cup_{i=o}^{\infty} F_{AF}^{i}(\emptyset)) \subseteq \texttt{Concs}(GE)$. □

To illustrate the applicability of our theory, we show that any argumentation framework obtained by the ASPIC$^-$ formalism (assuming a finite defeasible theory [3]) is c-superseded by a finitary argumentation framework. We refer to the finitary argumentation framework as a *finited* version of the original framework. Unfortunately, space restrictions prevent us from including all relevant definitions of ASPIC$^-$. For these, we refer the reader to [3] instead.

Proposition 3. *Let $\mathcal{A}rgs$ be an infinite set of arguments of a particular* ASPIC$^-$ *theory. There exists a rule r that has no upper bound in the number of times it can occur in the same branch of an argument in $\mathcal{A}rgs$.*

Proof. Suppose, towards a contradiction, that there exists an upper bound, say n. This means that each argument in the infinite set $\mathcal{A}rgs$ has each rule in the defeasible theory occurring at most n times in the same branch. This implies that the depth of each argument in $\mathcal{A}rgs$ is at most $n \cdot |\mathcal{R}|$. Let m be the size of the largest antecedent of the rules in $\mathcal{R}$ (that is, m is the biggest "fan-out" factor one can get when constructing an argument). Then the maximal number of rule-occurrences in each argument is $m^{n \cdot |\mathcal{R}|}$. Even if one takes into account all possible permutations of the rules in an argument, the result is still finite. But this means it is impossible to obtain an infinite number of arguments in $\mathcal{A}rgs$. □

Theorem 4. *Let $AF = (Ar, def)$ be generated by a finite* ASPIC$^-$ *theory. There exists a finitary argumentation framework $AF' = (Ar', def')$ that c-supersedes it.*

Proof. We distinguish two cases: weakest link and last link.

weakest link Assume that AF has been generated using Ewl or Dwl. We first observe that for each argument A with a same-branch repeating rule, there exists an argument A^* without any same-branch rule, such that A^* c-supersedes A. The idea is to construct this A^* by iteratively applying subargument substitution. Let A be an argument that has a same-branch repeating rule. That is, $\exists A_1, A_2 : A_1 \in \texttt{Sub}(A) \wedge A_2 \in \texttt{Sub}(A_1) \wedge \texttt{TopRule}(A_1) = \texttt{TopRule}(A_2)$. Substitute A_2 for A_1 in A. Keep on doing substitutions like this until there are no same-branch repeating rules anymore. Call the resulting argument A^*. At each substitution step, the argument after the step (say A'') c-supersedes the argument before the step (say A') for the following reasons.

1. $\texttt{Conc}(A'') = \texttt{Conc}(A')$
2. $A'^+ \subseteq A''^+$. Suppose A' defeats B. We distinguish two cases.
   - A' undercuts B. Then A'' also undercuts B (since $\texttt{Conc}(A'') = \texttt{Conc}(A')$)
   - A' rebuts B and $\texttt{DefRules}(A') \not\preceq_{\{\texttt{Ewl},\texttt{Dwl}\}} \texttt{DefRules}(B')$ (where B' is the subargument of B whose top-conclusion is defeated). Since $\texttt{Conc}(A'') = \texttt{Conc}(A')$ it follows that A'' rebuts B. Since $\texttt{DefRules}(A'') \subseteq \texttt{DefRules}(A')$ it follows that $\texttt{DefRules}(A'') \not\preceq_{\{\texttt{Ewl},\texttt{Dwl}\}} \texttt{DefRules}(B')$.
3. $A'^- \supseteq A''^-$. Suppose A'' is defeated by B. We distinguish two cases.
   - B undercuts A''. Since $\texttt{DefRules}(A'') \subseteq \texttt{DefRules}(A')$ it follows that B also undercuts A'.

- B rebuts A'' and $\texttt{DefRules}(B) \not\prec_{\{\texttt{Ewl},\texttt{Dwl}\}} \texttt{DefRules}(A''')$ (where A''' is the subargument of A'' whose top-conclusion is defeated). Since $\texttt{DefRules}(A_2) \subseteq \texttt{DefRules}(A_1)$ it follows that $\texttt{DefRules}(A''') \subseteq \texttt{DefRules}(A'''')$ (where A'''' is the subargument of A' whose top-conclusion is defeated). Hence, $\texttt{DefRules}(B) \not\prec_{\{\texttt{Ewl},\texttt{Dwl}\}} \texttt{DefRules}(A'''')$.

Let $AF' = (Ar', def')$ be the argumentation framework where Ar' consist of each A^* resulting from an $A \in Ar$ and def' be $def \cap (Ar' \times Ar')$. From the above, it follows that AF' c-supersedes AF. We now prove that AF' is finite. Suppose, towards a contradiction, that AF' is infinite. Proposition 3 tells us that there is a rule that has no upper bound in the number of times it can occur in the same branch. However, each argument $A^* \in Ar'$ has each rule occurring at most once in each branch. So there actually *is* an upper bound (it's 1). Contradition.

last link Assume that AF has been generated using `Ell` or `Dll`. We first observe that with last link, we cannot always carry out the same kind of substitutions as with weakest link and still expect the resulting argument to c-supersede the original argument. The reason is that we cannot be sure that $\texttt{LastDefRules}(A_2) \subseteq \texttt{LastDefRules}(A_1)$. It appears that an alternative strategy is needed. Instead of performing a substitution whenever there are two occurrences of the same rule in the same branch, we only perform substitution if, in addition, these two rule-occurrences also have the same `LastDefRules`. That is, let $A \in Ar$ be such that $\exists A_1, A_2 : A_1 \in \texttt{Sub}(A) \wedge A_2 \in \texttt{Sub}(A_1) \wedge \texttt{TopRule}(A_1) = \texttt{TopRule}(A_2) \wedge \texttt{LastDefRules}(A_1) = \texttt{LastDefRules}(A_2)$ then substitute A_2 for A_1 in A. Keep doing substitution steps like these until there are no same-branch repeated rules with the same `LastDefRules`. Call the resulting argument A^* and let $AF' = (Ar', def')$ be the associated argumentation framework. Following similar reasoning as for the weakest link case above, it follows that AF' c-supersedes AF.
We still have to prove that AF' is finite. This requires some additional effort, because now a rule can occur more than once in the same branch (as long as they have different `LastDefRules`). Suppose towards a contradiction that AF' is infinite. Then Proposition 3 tells that there is a rule that has no upper bound in the number of times it can occur in the same branch. But as the number of rules in the defeasible theory is finite, it follows that at some point `LastDefRules` will start to become the same (this is because there is only a finite number of subsets of $\mathcal{R}_d$ that can serve as `LastDefRules`). But this is impossible, because then this multiple rule-occurrence should have been substituted away during the substitution process. Contradiction. □

Theorem 5. *Let $AF = (Ar, def)$ be generated by a finited* ASPIC$^-$ *theory and let GE be the grounded extension of AF. It holds that* $\texttt{Concs}(GE) = \texttt{Concs}(\cup_{i=0}^{\infty} F_{AF}^i(\emptyset))$.

Proof. This follows directly from Theorem 3 and Theorem 4. □

What the above theorem shows is that if we want to compute the conclusions yielded by ASPIC$^-$ under grounded semantics, then we are free to do so using the iterative procedure, even though the argumentation framework generated by the ASPIC$^-$ theory might not be finitary.

5 Discussion and Conclusions

In this paper we formalise the concept of one argument superseding another. Since a finite set of rules can generate an infinite set of arguments, the results presented in this paper are of critical importance — they allow us to reduce such infinite frameworks into finite ones, and enable us to compute the grounded semantics over such frameworks in a standard way. While our results focused on the ASPIC$^-$ framework of [3], they are also directly applicable to other ASPIC style frameworks [1, 10, 9], as well as the argumentation interpretation of ABA [6] and argument-based Logic Programming [12, 4].

With regards to related work, [7] introduces a redundancy relation between arguments. The aim of [7] was to identify postulates necessary for generic argument systems to be useful. Redundancy was therefore used to *trim* large argument systems, obtained from formalisms such as ABA, into smaller systems which comply with their postulates. [7] showed that such *trimmed* frameworks (i.e. those without redundant arguments) yield the same extensions as untrimmed frameworks. Unlike the present work, [7] did not consider the validity of the inductive definition for the grounded semantics in the presence of infinitary argumentation systems. Furthermore, our results are applicable to instantiated

frameworks which make use of unrestricted rebut (such as ASPIC$^-$), and which are therefore arguably more natural to use for the reasoning about argument in real domains (see the discussion in [3]).

Another line of research where our results are relevant is in embedding classical logic into rule based formalisms. Approaches such as ASPIC-lite [13] and [8] seek to embed propositional logic into an ASPIC style system. Classical entailment can lead to an infinite number of attackers, for reasons other than reoccurring rules or propositions. For example, consider the defeasible rules $\Rightarrow a$, $\Rightarrow b$ and $\Rightarrow \neg(a \wedge b)$. When using strict rules as classical inference, the arguments $A_0 :\Rightarrow \neg(a \wedge b)$ has an infinite number of attackers, such as $B_1 : (\Rightarrow a), (\Rightarrow b) \rightarrow a \wedge b$; $B_2 : ((\Rightarrow a), (\Rightarrow b) \rightarrow a \wedge a \wedge b) \rightarrow a \wedge b$; $B_3 : ((\Rightarrow a), (\Rightarrow b) \rightarrow a \wedge a \wedge a \wedge b) \rightarrow a \wedge b$, etc. Our work can potentially be applied to show that the inductive definition of grounded semantics is still applicable in such situations.

Acknowledgements

The research in this paper was carried out within the project "Scrutable Autonomous Systems", funded by the Engineering and Physical Sciences Research Council, grant ref. EP/J012084/1.

References

[1] M.W.A. Caminada and L. Amgoud. On the evaluation of argumentation formalisms. *Artificial Intelligence*, 171(5-6):286–310, 2007.

[2] M.W.A. Caminada, R. Kutlak, N. Oren, and W.W. Vasconcelos. Scrutable plan enactment via argumentation and natural language generation. In *Proceedings AAMAS'14*, pages 1625–1626, 2014.

[3] M.W.A. Caminada, S. Modgil, and N. Oren. Preferences and unrestricted rebut. In Simon Parsons, Nir Oren, Chris Reed, and Frederico Cerutti, editors, *Computational Models of Argument; Proceedings of COMMA 2014*, pages 209–220. IOS Press, 2014.

[4] M.W.A. Caminada, S. Sá, and J. Alcântara. On the equivalence between logic programming semantics and argumentation semantics. In *Proceedings ECSQARU 2013*, pages 97–108, 2013.

[5] P.M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77:321–357, 1995.

[6] P.M. Dung, R.A. Kowalski, and F. Toni. Assumption-based argumentation. In Guillermo Simari and Iyad Rahwan, editors, *Argumentation in Artificial Intelligence*, pages 199–218. Springer US, 2009.

[7] P.M. Dung, F. Toni, and P. Mancarella. Some design guidelines for practical argumentation systems. In Pietro Baroni, Frederico Cerutti, Massimiliano Giacomin, and Guillermo R. Simari, editors, *Computational Models of Argument; Proceedings of COMMA 2010*, pages 183–194, 2010.

[8] D. Grooters and H. Prakken. Combining paraconsistent logic with argumentation. In Simon Parsons, Nir Oren, Chris Reed, and Frederico Cerutti, editors, *Computational Models of Argument; Proceedings of COMMA 2014*, pages 301–312. IOS Press, 2014.

[9] S. Modgil and H. Prakken. A general account of argumentation with preferences. *Artificial Intellligence*, 195:361–397, 2013.

[10] S. Modgil and H. Prakken. The ASPIC+ framework for structured argumentation: a tutorial. *Argument & Computation*, 5:31–62, 2014.

[11] E. Weydert. Semi-stable extensions for infinite frameworks. In Patrick de Causmaecker, Joris Maervoet, Tommy Messelis, Katja Verbeeck, and Tim Vermeulen, editors, *Proceedings of the 23rd Benelux Conference on Artificial Intelligence (BNAIC 2011)*, pages 336–343, 2011.

[12] Y. Wu, M.W.A. Caminada, and D.M. Gabbay. Complete extensions in argumentation coincide with 3-valued stable models in logic programming. *Studia Logica*, 93(1-2):383–403, 2009. Special issue: new ideas in argumentation theory.

[13] Y. Wu and M. Podlaszewski. Implementing crash-resistance and non-interference in logic-based argumentation. *Journal of Logic and Computation*, 2014. in print.

Capturing Evidence and Rationales with Requirements Engineering and Argumentation-Based Techniques

Marc van Zee [a] Sepideh Ghanavati [b]

[a] *University of Luxembourg, Luxembourg*
[b] *Public Research Center Henri Tudor, Luxembourg*

Abstract

User Requirements Notation (URN) is a modeling language with two complementary notations for capturing goals and non-functional requirements as well as low level processes and functional requirements. URN helps documenting the rationale behind the decisions, however, it lacks gathering the evidences for these rationales and documenting stakeholder discussions. In this paper, we propose a framework to extend URN with the Hybrid approach in reasoning based on evidences.

1 Introduction

Requirements Engineering (RE) has become an essential part of all development processes, especially in the area of complex reactive and distributed systems. The User Requirements Notation (URN) [11], which is based on ITU-T Standard, specifies visual notations to analyze functional (behaviour) and non-functional requirements (NFRs, such as performance, cost, security, and usability). URN consists of two complementary modeling notations: Goal Requirements Language (GRL) and Use Case Maps (UCMs). GRL aims at capturing business or system goals, alternative decisions to achieves high-level goals, and rationales behind goals and decisions. UCM tries to model functional and behaviorial requirements (requirements defining functions or behaviors of the system under development) using scenarios. A model containing GRL diagrams and UCMs is the result of an iterative refinement from both directions, based on discussions between stakeholders, who naturally give arguments for or against decisions, where these arguments can be motivated by evidence. Although URN offers traceability between GRL elements and UCM elements, it is not possible to trace back elements of the models to the actual discussions between stakeholder and the evidence that this was based on.

In this paper, we propose to extend URN to capture discussions between stakeholders based on evidence. We base our extension on the *hybrid approach* from evidential reasoning about the facts of a criminal case [4], in which arguments and narratives can be used in conjunction as well as interchangeably. This hybrid approach was shown to be natural way of modeling the process of proof, the iterative process of constructing, testing and justifying hypotheses in crime investigation and decision-making [7].

We show that parts of a GRL diagram can be traced back to arguments and evidence. Arguments can attack and support each other, which allowed the designer to reason about the impact of changing argument structures or evidence on the GRL diagrams and UCMs. Moreover, using the hybrid approach we are also able to reason about the plausibility, consistency and completeness of UCMs by asking so-called *critical questions*.

1.1 Related Work

Goal modeling notations such as Goal-oriented Requirements Language (GRL) [3] together with Use Case Maps (UCM) [17] have been used to help decision makers analyze alternatives, capture rationale

behind the decisions [18, 13]. Gross and Yu [10] explored the use of goal-oriented approaches to provide links between business goals, architectural design decisions and structure in a systematic way.

These notations evaluate the satisfaction level of the high-level, fuzzy goals of the organization via selecting a set of alternatives at lower levels. These analysis mainly focus on qualitative data range. However, Akhigbe et al. [9] introduce a method to be able to quantify the analysis evaluations for goals models. They propose the use of AHP technique in EA to perform the analysis techniques quantitatively. Furthermore, the Business Intelligence - Enabled Adaptive Enterprise Architecture (BI-EAEA) framework [1] has been proposed to use BI theme to help decision making process in EA as well as to support organizational business objectives alignements to daily operations and to provide a coherent view across the domains of the EA. Issue-Based Information System (IBIS) was invented by Werner Kunz and Horst Rittel as an argumentation-based approach to solving wicked problems, which are complex, ill-defined problems that involve multiple stakeholders [12]. Recently, there has been a renewed interest in IBIS-type systems, particularly in the context of sensemaking and collaborative problem solving in a different social and technical contexts. Our approach is comparable to IBIS, except that we make an explicit connection between arguments and stories, while IBIS mostly focuses on arguments.

The rest of this paper is organised as follows. In Section 2 we introduce our running example. In Section 3, we introduce the User Requirements Notation (URN), together with its sub-notations, GRL and UCM scenario notations, followed by a discussion on existing traceability within the URN. In Section 4, we introduce the formal theory of story schemes and arguments using in reasoning about facts of a crime and we also show how we can model a UCM scenario as a story scheme with arguments, and we demonstrate that it leads to an increase of traceability using our running example.

2 Running Example

The stakeholders of a fictitious company "Best Furniture Inc." have agreed through several meetings that they should improve customer support. The reason behind this is logical: Improving customer support leads to more profit because it will in the long term result in more customer, and more customers leads to more profit. It is clear that increasing profit is one of the main goal of any company.

In order to implement this goal, they have decided that customers should be able to return a product when they are not satisfied with it. The stakeholders initially agree that the following three conditions that should be satisfied if a customer would like to return a product:

1. The product is bought from company "Best Furniture Inc.".
2. The customer has a receipt for the product.
3. The product is not damaged.

Condition (1) and (2) are separated due to the rationale that every product of the company "Best Furniture Inc." is registered in their system, thus they are able to check whether the product is bought without a receipt. The rationale for condition (3) is clear: The company is unable to resell damaged products.

After some more discussion, the stakeholders decide that it would be useful to make a distinction between condition (1) and (2), and they decide that if a customer does not have a receipt, he or she is still entitled to company store credit. The stakeholders believe that this will increase their customer support even more. Their statistics also show that the number of customers trying to return a product without a receipt is relatively low, thus it will not cost them much.

3 The User Requirements Notation (URN)

User Requirements Notation (URN) [11], an ITU-T Standard, is one of the first modeling languages in the area of RE which aims at the standardization of visual notations to analyze functional (behaviour) and non-functional requirements (NFRs) such as performance, cost, security, and usability. URN allows software and requirements engineers to identify requirements for a proposed or an evolving system and to review such requirements for correctness and completeness. URN combines two complementary notations: the *Goal-oriented Requirement Language* (GRL) [3] for goals and NFRs, and *Use Case Maps* (UCMs) [17] for scenarios, business processes and functional requirements. GRL, which is based on

some existing goal modeling notations (i.e. *i** and NFR Framework), captures business or system goals, alternative ways of satisfying goals and rationales for alternatives, their impact and the decisions. UCM helps modeling functional requirements, business processes and architecture structures using scenarios.

URN supports the reasoning about scenarios by establishing links between intentional elements (such as softgoals and goals) in GRL and non-intentional elements in UCM. Modeling both goals and scenarios is complementary and may aid in identifying further goals and additional scenarios (and scenario steps) important to stakeholders, thus contributing to the completeness and accuracy of the requirements.

The Use Case Map (UCMs) scenario notation focuses on causal relationships between activities or processes of one or more use cases. The relationships are said to be causal due to their involvement in concurrency and partial orderings of responsibilities, their links between causes and effects, and the level of abstraction from component interactions expressed as message exchanges. UCMs are applicable to use case capturing and elicitation, use case validation, as well as high-level architectural design and test case generation. They provide a behavioural framework for evaluating and making architectural decisions at a high level of design.

URN has an open source Eclipse-based tool support called jUCMNav [14]. jUCMNav supports modeling UCM, GRL and the *URN links*(▶) between them. jUCMNav provides the support for multiple view of the same model to help scalability. The tool prevents the creation of syntactically incorrect URN models through hard-coded rules, and more specific styles can be enforced through user-selectable semantic rules written in Object Constraint Language. jUCMNav supports several GRL evaluation algorithms such as quantitative, qualitative, hybrid, constraint-based or conditional evaluation algorithms. However, there is only one UCM path traversal scenario mechanism (with a few parameters). It also allows exporting individual or complete models to different graphical and/or report formats and exporting the results of GRL strategy evaluations in a comma-separated value (CSV) files.

3.1 Basics of GRL Notation

GRL syntax is based on *i** language which describes business concerns such as low cost, fast time to market, high customer satisfaction as well stakeholders' beliefs and dependencies. *Softgoals*(), are goals which are somewhat fuzzy in nature and cannot be entirely satisfied. *Softgoals* capture high-level objectives. *Goals*() represent goals that are not fuzzy and can be clearly achieved. *Tasks* (), are typically used to operationalize parent (soft-)goals and/or goals. Goals, softgoals and tasks can be connected to each other, in an AND/OR/XOR graph, using *decomposition* links(), or can be connected with *contribution*() or *correlation*() links showing the impact on each other. Contribution or correlation links may have various degrees of impact qualitatively, including necessary and sufficient (*make*), necessary but insufficient (*help*), unknown positive (*some+*), and their corresponding contributions on the negative side (*break, hurt*, and *some-*). Contribution and correlation links have quantitative counterparts, which range from -100 to 100. Justifications or explanations of GRL intentional elements and links can be added in the form of *beliefs* (). Beliefs document the rationale behind various parts of a GRL model. Stakeholders can support or argue against such beliefs while validating the model. Beliefs provide a means to prevent stakeholders from having the same discussion over and over again. Although we will only discuss this simple example due to space constraints, GRL models contain much more components. See [2] for a more detailed overview.

Consider the GRL diagram in Figure 1, which is the resulting goal model of our running example. Both the goal Return Product for Money and Return Product for Company Credit contribute positively to the softgoal Improve Customer Support. The goal Return Product is decomposed into three subgoals Bought Item, Have Receipt, and Not Damaged. These subgoals are materialized with the tasks Check if the Item is Bought, Check for Receipt, and Check for Damage. Similarly, the goal Return Products for Company Credit has two subgoals which are materialized by two tasks. The rationalization of the different elements is provided by attaching belief elements to corresponding GRL elements.

3.2 Basic UCM Notation

In UCMs, a *scenario* is a partial description of the system usage defined as a set of partially-ordered responsibilities a system performs to transform inputs to outputs while satisfying preconditions and post-

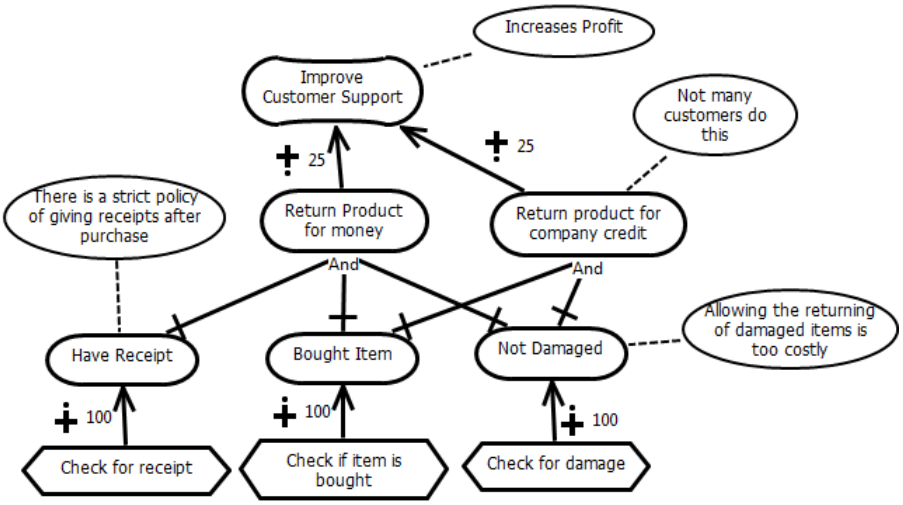


Figure 1: GRL Model for Customer Support

conditions. UCM *responsibilities*(×) are scenario activities representing something to be performed or some tasks to be achieved. A responsibility can potentially be associated or allocated to a *component*(▫). In UCMs, a component is generic and abstract enough to represent software entities (e.g. objects, processes, databases, or servers) as well as non-software entities (e.g. actors, agents or hardware).

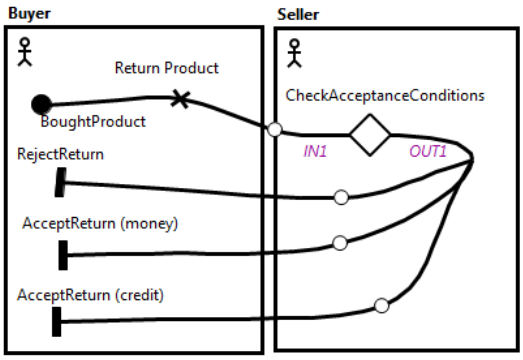


Figure 2: UCM Model for Product Return Procedure

The UCM in Fig 2 illustrates some of these concepts through a description of the product return procedure from our running example. Filled circles represent *start points*(●), which capture preconditions and triggering events. *End points*(▮) capturing resulting events and postconditions are illustrated with bars perpendicular to causal *paths*. Responsibilities can be assigned to *components* such as agents. In this example, there are two agents, namely the buyer and the seller. Scenarios progress along paths from start points to end points. Paths also include responsibilities. Paths can fork as alternatives (*OR-fork*(⌒)) and may also join (*OR-join*) or they can show concurrency with *AND-fork* (⊣⊢) and *AND-join*. Alternative branches can be guarded by *conditions*, shown between square brackets. A condition needs to be true for the guarded path to be followed.

UCM also contains a diamond symbol called a *stub*(◇), which is used as a container for a sub-processes. The stub can be dynamic or static. Dynamic stubs have more than one sub-maps whereas static stubs have only one sub-map. These sub-processes are called *plug-in* maps (since they are plugged into a stub). The plug-in map for the stub CheckAcceptanceConditions is depicted in Figure 3. Stubs have identifiable input and output segments ($IN1, OUT1, \ldots$) connected to start points and end points in the plug-in. This *binding relationship* ensures that paths flow from parent maps to sub-maps, and back to parent maps.

In this example, the return conditions are twice represented as an OR-fork, allowing the possibility

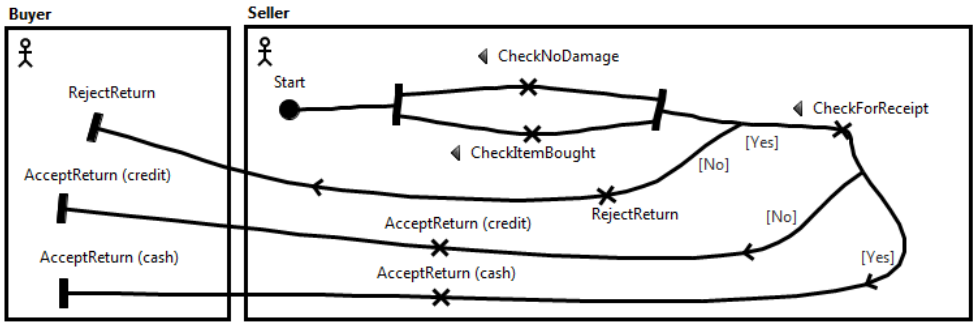


Figure 3: UCM Model to Check Acceptance Conditions

to reject or accept the return, depending on the result of the responsibilities. In the first case, the two responsibilities CheckForItemBought and CheckForReceipt are in an AND-fork, and if one of them is false, the return will be rejected. In the second case, if the responsibility CheckForDamage is true, the return is accepted.

3.3 Traceability in URN

The various GRL elements can be connected to UCMs in several of ways. Tasks in GRL can be linked to responsibility in UCM, or actors in GRL can be mapped to components in UCM. A UCM could also elaborate a collection of tasks (for instance, to explore ordering among these tasks) or be connected directly to goals or softgoals in the GRL model. For instance, the responsibilities CheckItemBought, CheckForReceipt, and CheckNoDamage in the UCM in Figure 3 are linked to the goals Bought Item, Have Receipt, and Not Damaged of the GRL in Figure 1, respectively. Also, the goals Return Product for Cash and Return Product for Credit are linked to the entire UCM in Figure 2. Such traceability relationships are important, especially during the evolutions of the system where they can be used for impact analysis.

A goal not covered by any scenario is a symptom of an incorrect or overspecified GRL models or of an incomplete UCM model. Similarly, a scenario that does not contribute to any goal is either not necessary, or the goal model needs to be enhanced.

4 Traceability of Arguments and Evidence in URN

As we have seen in the previous section, URN offers some traceability of UCM elements to GRL elements through URN links. This allows the user to verify the completeness and consistency of both the GRL diagrams and the UCM scenarios by checking for "gaps". This also allows stakeholders to iteratively improve their model by working on both sides and constantly check whether there is still traceability between the two approaches.

URN provides some traceability of rationalization for GRL elements as well, namely using Belief elements. Beliefs can be seen as *arguments* for certain elements, which are the result of *discussions* between the stakeholders. Still, it seems that capturing a discussion using a single belief element is rather limited. This is also clearly seen from our running example. The discussion process between the stakeholders described in Section 2 shows that initially, the stakeholders agreed that they should not allow customers to return products when they do not have a receipt, but this argument was later defeated by another argument, namely that allowing customers to return products without a receipt for company credit leads to a bigger increase in customer service, and by the evidence that this does not occur very often.

The GRL language is not able to model such argument structures, because the beliefs elements in GRL are *flat*. There do not exist attack or support relations between belief elements, and beliefs cannot be supported by evidence either. Therefore, the actual discussion that occurred between stakeholders is lost, meaning that there is no traceability from these discussions to the resulting GRL elements. In this section, we propose a framework that extends URN with such argument structures. Our approach is based on the so-called *hybrid theory* used in reasoning about evidence in criminal cases [6].

4.1 The Hybrid Approach

Two main approaches exists in the literature on reasoning on the basis of evidence, namely the argument-based approach and the narrative-based approach. Recently, Bex et al. [5] have introduced a *hybrid approach*, in which arguments and narratives can be combined with each other in a single theory. This approach turned out to be a natrual way of modeling the process of proof, the iterative process of constructing, testing and justifying hypotheses in crime investigation and decision-making [7]. Moreover, it was used as the starting point for a sense-making tool, AVERS [8], in which the knowledge about a case can be mapped and visualized using both stories and arguments. We will now show how we can model our running example using the hybrid approach, and that it fits the URN specification very well. We will discuss the argument-based side of the hybrid approach, and next the narrative-based side. Finally, we discuss how to combine them.

In Figure 4, an example of an argument tree from the argument-based approach is given, which reflects part of the state of the discussion between the stakeholders in our running example of Section 2. The arrows between the premises and (intermediate) conclusions specify (defeasible) inferences. The coloured boxes (leaves of the tree) are *evidence* for the different arguments. This evidence can be an opinion of a group of stakeholders, or more formal evidence such as a document or a scientific article. For instance, evidence E1 can be a statement that all stakeholders have agreed upon such as the mission statement of the company. A defeasible inference is associated with an underlying generalization that acts as a *warrant* [16]. For example, the inference from evidence E1 to the argument "We want to increase profit" is justified by the generalization 'When something is stated in a mission statement of the company, the arguments holds', which can be rephrased as 'the mission statement mentions p, therefore (presumable) p'.

We can observe that this argument tree has links with GRL elements, which provides the means to trace back GRL elements to the argument tree. For instance, the GRL goal Return Product for money of Figure 1 can be traced back to the top argument, while the GRL softgoal Improve Customer Support can be traced back to the sub-argument "We want to increase customer support".

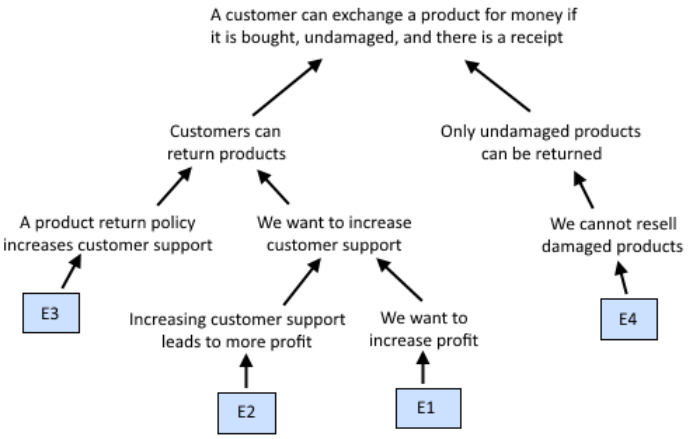


Figure 4: An argument based on evidence

Although Figure 4 only shows arguments that support each other, arguments can also be *attacked* by other arguments. They can be *rebutted* with an argument for the opposite conclusion and they can be *undercut* with an argument for why an inference is not allowed (this is mostly because some generalization does not apply in the given circumstances) [7]. In the example, an argument for "Customers cannot return products" rebuts the argument "Customers can return products" and an argument for "The mission statement is interpreted incorrectly" undercuts the inference from E1 to "We want to increase profit". These attacking arguments can again be attacked and thus the status of arguments (e.g. "justified", "overruled") can be determined dialectically.

Now that we have explained how we can integrate the argument-based side of the hybrid approach into URN, we will continue with the narrative-based side. The narrative-based approach used in reasoning about evidence in a criminal case is based on the observation that people tend to organize evidence by building stories about what might have happened [15]. Such stories are essentially network of events

which are causally linked. This is very similar to the way that UCMs are built up (Figure 2 and 3). UCMs contain causally linked events that are called responsibilities. Therefore, the narrative-based side of the hybrid approach can directly be integrated into URN, simply by treating the UCMs as narratives.

The main idea of the hybrid approach is to link arguments in the argument-based approach with events in narrative-based approach. We can do the same thing for URN by treating the narratives as the UCMs. As we have explained above, argument trees have several points in which they can be linked with GRL elements. Moreover, GRL elements in turn have traceability links with URN elements as well (provided by URN, see Section 3.3). Thus, GRL elements provide the connection to trace UCM elements back to the arguments between the stakeholders and the underlying evidence.

5 Discussion

Having traceability from UCM elements to discussions between the stakeholders and the underlying evidence is useful for several purposes. First of all, URN elements can be firmly anchored or, in other terms, *evidentially supported*. As we mentioned in Section 3.3, the entire UCM of Figure 2 can be traced back to the goals Return Product for Money and Return Product for Company Credit, which in turn can be traced back to the argument "Customers can return products", which is based on several sub-arguments and evidence. Arguments can be attacked, which may break the "anchor's chain", causing the story to be no longer connected to the ground [7]. If, for example, the argument "Customers can return products" is defeated by another argument, then the corresponding UCM of Figure 2 is no longer supported by an argument. Aside from the anchoring stories in evidence, traceability from arguments between stakeholders to URNs also makes it possible to reason about the coherence of a URN model in a dialectical way.

Since our approach is very similar to the hybrid approach, we can use two types of criteria that are used in the hybrid approach to determine the quality of a narrative to determine the quality of an UCM. The first one is the extent to which a UCM conforms to the evidence (anchoring), and the second one is the coherence (consistency of causal relations). Bex [4] gives these and other criteria as a list of *critical questions*, typical sources of doubt when reasoning with the hybrid theory. The questions that concern the extend to which the story conforms to the evidence are as follows:

- *Evidential support* or *anchoring*: How much and which of the available evidence supports the story?
- *Evidential contradiction:* How much and which of the available evidence contradicts the story?
- *Evidential gaps:* How many and which events in the explanation are unsupported by evidential data?

The other three questions concern a story's coherence, the extent to which a story conforms to commonsense knowledge:

- *Plausibility:* How plausible are the events and causal relations in the story?
- *Completeness:* Does the story adhere to a plausible story scheme?
- *Consistency:* Are there elements of the story that contradict each other?

It is clear that we can almost directly translate these questions into our framework, with the exception of the question about completeness. This question mentions so-called *story schemes*, but we will not discuss them here.

6 Conclusion

In this paper, we discussed and analyzed URN modeling language for capturing rationale and design decision through a simple example. We identified the need for documenting evidences to help argumentation against or for rationales. Thus, we proposed to extend URN with a hybrid approach which helps in identifying evidence with both argument-based and narrative approaches.

7 Acknowledgments

This work has been partially funded by FNR PEARL programme and AFR - PDR grant #5810263, Luxembourg.

References

[1] Okhaide Akhigbe, Daniel Amyot, and Gregory Richards. A framework for a business intelligence-enabled adaptive enterprise architecture. In *33rd Int. Conf. on Conceptual Modeling (ER'014)*, LNCS, 2014.

[2] Daniel Amyot. Introduction to the user requirements notation: Learning by example. *Comput. Netw.*, 42(3):285–301, June 2003.

[3] Daniel Amyot, Sepideh Ghanavati, Jennifer Horkoff, Gunter Mussbacher, Liam Peyton, and Eric S. K. Yu. Evaluating goal models within the goal-oriented requirement language. *Int. J. Intell. Syst.*, 25:841–877, August 2010.

[4] Floris Bex. Analysing stories using schemes. *Legal Evidence and Proof: Statistics, Stories, Logic*, pages 93–116, 2009.

[5] Floris J. Bex, Henry Prakken, and Bart Verheij. Formalising argumentative story-based analysis of evidence. In *ICAIL*, pages 1–10. ACM, 2007.

[6] Floris J. Bex and SpringerLink (Online service). *Arguments, Stories and Criminal Evidence*. Law and Philosophy Library,.

[7] Floris Jurriaan Bex and Bart Verheij. Story schemes for argumentation about the facts of a crime. In *AAAI Fall Symposium: Computational Models of Narrative*, volume FS-10-04 of *AAAI Technical Report*. AAAI, 2010.

[8] S.W. van den Braak. *Sensemaking Software for Crime Analysis*. PhD thesis, Utrecht, The Netherlands, 2010.

[9] Okhaide Akhigbe et al. Creating quantitative goal models: Governmental experience. In *33rd Int. Conf. on Conceptual Modeling (ER'014)*, LNCS, 2014.

[10] D. Gross and E. Yu. Evolving system architecture to meet changing business goals: an agent and goal-oriented approach. In *Proc. 5th IEEE Int. Symposium on RE*, pages 316–317, 2001.

[11] ITU-T. Recommendation Z.151 (11/08): User Requirements Notation (URN) – Language Definition. `http://www.itu.int/rec/T-REC-Z.151/en`, 2008.

[12] Werner Kunz, Horst W. J. Rittel, We Messrs, H. Dehlinger, T. Mann, and J. J. Protzen. Issues as elements of information systems. Technical report, 1970.

[13] Lin Liu and Eric Yu. Designing web-based systems in social context: A goal and scenario based approach. In *Advanced Information Systems Engineering*, volume 2348 of *LNCS*, pages 37–51. Springer Berlin Heidelberg, 2002.

[14] Gunter Mussbacher and Daniel Amyot. Goal and scenario modeling, analysis, and transformation with jUCMNav. In *ICSE Companion*, pages 431–432, 2009.

[15] Nancy Pennington and Reid Hastie. Reasoning in explanation-based decision making. *Cognition*, 49(12):123 – 163, 1993.

[16] Stephen E. Toulmin. *The Uses of Argument*. Cambridge University Press, July 2003.

[17] M. Weiss and D. Amyot. Designing and evolving business models with URN. In *Proc. of Montreal Conference on eTechnologies (MCETECH05)*, pages 149–162, 2005.

[18] E. Yu, M. Strohmaier, and Xiaoxue Deng. Exploring intentional modeling and analysis for enterprise architecture. In *10th IEEE EDOCW '06. Int.*, pages 32–32, 2006.

Fast Laplace Approximation for Gaussian Processes with a Tensor Product Kernel

Perry Groot [a] Markus Peters [b] Tom Heskes [a] Wolfgang Ketter [b]

[a] *Radboud University Nijmegen*
[b] *Erasmus University Rotterdam*

Abstract

Gaussian processes provide a principled Bayesian framework, but direct implementations are restricted to small data sets due to the cubic time cost in the data size. In case the kernel function is expressible as a tensor product kernel and input data lies on a multidimensional grid it has been shown that the computational cost for Gaussian process regression can be reduced considerably. Tensor product kernels have mainly been used in regression with a Gaussian observation model since key steps in their algorithms do not easily translate to other tasks. In this paper we show how to obtain a scalable Gaussian process framework for gridded inputs and non-Gaussian observation models that factorize over cases. We empirically validate our approach on a binary classification problem and our results shows a major performance improvement in terms of run time.

1 Introduction

Gaussian processes (GPs) provide a rich, principled, and well-established Bayesian framework for tasks such as non-linear non-parametric regression and classification [8]. A direct implementation of GPs, however, limits their applicability to small data sets since the kernel matrix needs to be stored, costing $\mathcal{O}(N^2)$, and inverted, costing $\mathcal{O}(N^3)$, with N the number of data points. In recent years, several approaches have addressed this problem often selecting a subset of the training data (the active set) of size M reducing the computational complexity to $\mathcal{O}(M^2N)$ for $M \ll N$ [6, 10]. An alternative approach, is to exploit additional structure in the GP model to reduce the computational complexity.

The structural assumption that we exploit in this paper is the assumption of input data lying on a multidimensional grid.[1] Such data is often found in regression applications in time and space, for example, regular measurements over time and space from weather stations. This assumption allows the kernel function to be written as a tensor product kernel that allows for efficient computations using Kronecker products [5]. In recent years, the use of multidimensional grids has attracted increasing attention, and has been used in a variety of applications such as image reconstruction [3, 14], point process intensity estimation [2], network reconstruction [11], and density estimation [9].

The Kronecker product has, however, been limited to GP regression with spherical noise. The main reason for this is that a critical step is to compute the inverse $(\boldsymbol{K} + \sigma^2\boldsymbol{I})^{-1}$, which is not a Kronecker product even if $\boldsymbol{K}$ is. In this particular case it is still possible to efficiently compute the inverse using an eigen decomposition $\boldsymbol{K} = \boldsymbol{Q}\boldsymbol{\Lambda}\boldsymbol{Q}^T$ and the identity $(\boldsymbol{K} + \sigma^2\boldsymbol{I})^{-1}\boldsymbol{y} = \boldsymbol{Q}(\boldsymbol{\Lambda} + \sigma^2\boldsymbol{I})^{-1}\boldsymbol{Q}^T\boldsymbol{y}$. This identity, however, no longer holds with non-spherical noise and how to efficiently obtain an eigen decomposition in this case is in fact an open problem [4].

In this paper we propose a scalable GP framework for gridded inputs and non-Gaussian observation models that factorize over cases. Section 2 briefly introduces GPs, the Laplace approximation, and the Kronecker product. Thereafter we describe how to efficiently combine tensor product kernel functions in the Laplace approximation. We provide derivations and pseudocode for obtaining the mode and for hyperparameter learning in Sections 3 and 4. Section 5 reviews the memory and run time requirements of the algorithms. In Section 6 we empirically validate the new methods and Section 7 gives conclusions.

[1] The grid does not necessarily have to be equispaced.

2 Background

We denote vectors $\boldsymbol{x}$ and matrices $\boldsymbol{K}$ with bold-face type and their components with regular type, i.e., x_i, K_{ij}. With $\boldsymbol{x}^T$ we denote the transpose of the vector $\boldsymbol{x}$. Let $\boldsymbol{x} \in \mathbb{R}^D$ be an input, $\boldsymbol{y} \in \mathcal{Y}$ an output. We denote with $\boldsymbol{X}, \boldsymbol{Y}$ the observed data. We denote with $|\boldsymbol{A}|$ the determinant of $\boldsymbol{A}$, $\boldsymbol{A} \otimes \boldsymbol{B}$ the Kronecker product, $\boldsymbol{A} \circ \boldsymbol{B}$ the Hadamard (element wise) product, and vec$(\boldsymbol{A})$ the vector obtained by stacking all the columns of matrix $\boldsymbol{A}$. Given a vector $\boldsymbol{a}$ and matrix $\boldsymbol{A}$, diag$(\boldsymbol{a})$ gives a diagonal matrix with $\boldsymbol{a}$ on its diagonal, and diag$(\boldsymbol{A})$ gives the vector of all diagonal elements of $\boldsymbol{A}$. Hyperparameters are denoted by $\boldsymbol{\theta} = \{\boldsymbol{\theta}^c, \boldsymbol{\theta}^l\}$, with $\boldsymbol{\theta}^c$ the kernel hyperparameters and $\boldsymbol{\theta}^l$ the likelihood hyperparameters.

2.1 Gaussian Processes

A Gaussian process (GP) is a collection of random variables $\{f(\boldsymbol{x}_i)\}_{i \in I}$ for some index set I, any finite number of which have a joint Gaussian distribution [8]. A GP $f \sim \mathcal{N}(\boldsymbol{m}, \boldsymbol{K})$ results in a finite multivariate Gaussian distribution where each element of the covariance matrix $K_{ij} = k(\boldsymbol{x}_i, \boldsymbol{x}_j)$ is given by a kernel function $k(\cdot,\cdot)_{\boldsymbol{\theta}^c}$ with parameters $\boldsymbol{\theta}^c$. A GP effectively specifies a prior distribution over functions $f(\cdot)$ in a Bayesian framework in which the likelihood model $p(y_i|f(\boldsymbol{x}_i))$ is parameterized by $f(\cdot)$. Given a GP prior and likelihood, Bayes formula gives us the posterior distribution $p(\boldsymbol{f}|\boldsymbol{Y}, \boldsymbol{X}, \boldsymbol{\theta}) = p(\boldsymbol{Y}|\boldsymbol{f}, \boldsymbol{\theta}_l)p(\boldsymbol{f}|\boldsymbol{X}, \boldsymbol{\theta}_c)/p(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{\theta})$. In the simplest case, observations are assumed to be generated by an unknown function possibly corrupted with Gaussian noise, i.e., $y_i = f(\boldsymbol{x}_i) + \epsilon_i$ with $\epsilon_i \sim \mathcal{N}(0, \sigma^2)$ which leads to a model in which inference is analytically tractable. In this case the predictive distribution for a test location $\boldsymbol{x}_*$ is given by $f_*|\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{x}_* \sim \mathcal{N}(\mu_*, v_*)$ with $\mu_* = k(\boldsymbol{x}_*, \boldsymbol{X})(\boldsymbol{K} + \sigma^2\boldsymbol{I})^{-1}\boldsymbol{Y}$ and $v_* = k(\boldsymbol{x}_*, \boldsymbol{x}_*) - k(\boldsymbol{x}_*, \boldsymbol{X})(\boldsymbol{K} + \sigma^2\boldsymbol{I})^{-1}k(\boldsymbol{X}, \boldsymbol{x}_*)$. For non-Gaussian likelihoods one needs to resort to sampling methods or approximations.

2.2 Laplace Approximation

With a non-Gaussian likelihood inference in Gaussian process models leads to analytically intractable integrals when making predictions. For example, the distribution of a latent variable given a test case is given by

$$p(\boldsymbol{f}_*|\boldsymbol{X}, \boldsymbol{y}, \boldsymbol{x}_*) = \int p(f_*|\boldsymbol{X}, \boldsymbol{x}_*, \boldsymbol{f})p(\boldsymbol{f}|\boldsymbol{X}, \boldsymbol{y})d\boldsymbol{f} \tag{1}$$

The Laplace approximation [13] gives an analytic approximation to these integrals by approximating the true posterior distribution $p(\boldsymbol{f}|\boldsymbol{X}, \boldsymbol{y})$ with a Gaussian $q(\boldsymbol{f})$ centered on the mode of the posterior. The variance is obtained through a second-order Taylor expansion around the posterior mode $\hat{\mathbf{f}}$, which results in

$$q(\boldsymbol{f}) = \mathcal{N}(\boldsymbol{f}|\hat{\mathbf{f}}, (\boldsymbol{K}^{-1} + \boldsymbol{W})^{-1}) \tag{2}$$

with $\hat{\mathbf{f}} = \arg\max_{\boldsymbol{f}} p(\boldsymbol{f}|\boldsymbol{X}, \boldsymbol{y}, \boldsymbol{\theta})$ and $\boldsymbol{W} = -\nabla\nabla_{\boldsymbol{f}} \log p(\boldsymbol{y}|\boldsymbol{f})|_{\boldsymbol{f}=\hat{\mathbf{f}}}$ which is diagonal since the likelihood factorizes over cases.

2.3 Kronecker Products

In this paper we assume a tensor product kernel function, i.e., $k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \prod_{d=1}^{D} k_d(\boldsymbol{x}_i^d, \boldsymbol{x}_j^d)$. Assuming a product kernel function together with input data on a multidimensional grid leads to a kernel matrix that decomposes into a Kronecker product of matrices of lower dimensions. The Kronecker product has a convenient algebra [5]:

$$\boldsymbol{A} \otimes \boldsymbol{B} = \begin{bmatrix} a_{11}\boldsymbol{B} & \cdots & a_{1n}\boldsymbol{B} \\ \vdots & \ddots & \vdots \\ a_{m1}\boldsymbol{B} & \cdots & a_{mn}\boldsymbol{B} \end{bmatrix} \qquad \begin{aligned} (\boldsymbol{A} \otimes \boldsymbol{B})\text{vec}(\boldsymbol{X}) &= \text{vec}(\boldsymbol{B}\boldsymbol{X}\boldsymbol{A}^T) \\ \boldsymbol{A}\boldsymbol{B} \otimes \boldsymbol{C}\boldsymbol{D} &= (\boldsymbol{A} \otimes \boldsymbol{C})(\boldsymbol{B} \otimes \boldsymbol{D}) \\ (\boldsymbol{A} \otimes \boldsymbol{B})^{-1} &= \boldsymbol{A}^{-1} \otimes \boldsymbol{B}^{-1} \end{aligned} \tag{3}$$

Algorithm 1 Laplace mode finding

1: **function** LAPLACEMODE(covariance matrix $\boldsymbol{K}$, observations $\boldsymbol{y}$, likelihood function $p(\boldsymbol{y}|\boldsymbol{f})$)
2: $\boldsymbol{f} \leftarrow \boldsymbol{0}$ ▷ Initialization
3: **repeat** ▷ Newton iteration
4: $\boldsymbol{W} \leftarrow -\nabla\nabla \log p(\boldsymbol{y}|\boldsymbol{f})$
5: $\boldsymbol{b} \leftarrow \boldsymbol{W}\boldsymbol{f} + \nabla \log p(\boldsymbol{y}|\boldsymbol{f})$
6: Iteratively solve: $(\boldsymbol{I} + \boldsymbol{W}^{\frac{1}{2}}\boldsymbol{K}\boldsymbol{W}^{\frac{1}{2}})\boldsymbol{v} = \boldsymbol{W}^{\frac{1}{2}}\boldsymbol{K}\boldsymbol{b}$ ▷ Eq. (4) using Conjugate Gradient
7: $\boldsymbol{a} \leftarrow \boldsymbol{b} - \boldsymbol{W}^{\frac{1}{2}}\boldsymbol{v}$
8: $\boldsymbol{f} \leftarrow \boldsymbol{K}\boldsymbol{a}$
9: **until** convergence ▷ Objective $-\frac{1}{2}\boldsymbol{a}^T\boldsymbol{f} + \log p(\boldsymbol{y}|\boldsymbol{f})$
10: **return** $\hat{\mathbf{f}} \leftarrow \boldsymbol{f}$ ▷ Mode
11: **end function**

The Kronecker product can be used to construct memory and computationally efficient algorithms. For example, for N data points a full $N \times N$ matrix needs $\mathcal{O}(N^2)$ memory storage and a matrix vector product is of order $\mathcal{O}(N^2)$ whereas for a Kronecker matrix in which each dimension d has cardinality N_d, this is $\mathcal{O}(\sum_{d=1}^{D} N_d^2)$ and $\mathcal{O}(N(\sum_{d=1}^{D} N_d))$, respectively [14]. Furthermore, the Cholesky and singular value decompositions both reduce from $\mathcal{O}(N^3)$ to $\mathcal{O}(\sum_{d=1}^{D} N_d^3)$ since these operations need only be applied to each component.

3 Maximum a Posteriori Estimation

The Laplace approximation approximates the posterior by a Gaussian centered on the mode of the posterior. The mode $\hat{\mathbf{f}}$ of the posterior is the maximizer of the log posterior, which can be found by setting the first derivative of the log posterior to zero and solving for $\hat{\mathbf{f}}$. We can find the maximum iteratively by using Newton's algorithm for which we need to compute in each iteration the following update step:

$$\begin{aligned} \boldsymbol{f}^{new} &= (\boldsymbol{K}^{-1} + \boldsymbol{W})^{-1}\boldsymbol{b} \\ &= \boldsymbol{K}(\boldsymbol{I} - \boldsymbol{W}^{\frac{1}{2}}(\boldsymbol{I} + \boldsymbol{W}^{\frac{1}{2}}\boldsymbol{K}\boldsymbol{W}^{\frac{1}{2}})^{-1}\boldsymbol{W}^{\frac{1}{2}}\boldsymbol{K})\boldsymbol{b} \end{aligned} \tag{4}$$

with $\boldsymbol{b} = \boldsymbol{W}\boldsymbol{f} + \nabla \log p(\boldsymbol{y}|\boldsymbol{f})$. In the last line, we used the matrix inversion lemma to rewrite the Newton iteration in a numerically advantageous way. Conjugate Gradient (CG) can now be used iteratively to solve the linear system $(\boldsymbol{I} + \boldsymbol{W}^{\frac{1}{2}}\boldsymbol{K}\boldsymbol{W}^{\frac{1}{2}})\boldsymbol{v} = \boldsymbol{W}^{\frac{1}{2}}\boldsymbol{K}\boldsymbol{b}$, avoiding the need to directly compute any expensive matrix inverses. The mode is then found by setting $\boldsymbol{a} = \boldsymbol{b} - \boldsymbol{W}^{\frac{1}{2}}\boldsymbol{v}$ and $\boldsymbol{f} = \boldsymbol{K}\boldsymbol{a}$. Pseudocode is given in Algorithm 1. In line 6 one can obtain an exact answer for a quadratic system if the iterated solver is run for N steps, however, reasonable results are usually already obtained in a small fraction of iterations. Since Kronecker matrix vector products are efficient this greatly reduces the cost of mode finding compared to a standard Cholesky implementation.

4 Model Selection

After obtaining the mode $\hat{\mathbf{f}}$ for a given hyperparameter $\boldsymbol{\theta}$, the next problem is to learn the value of $\boldsymbol{\theta}$. This can be done by minimizing the negative marginal likelihood

$$-\log p(\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{\theta}) \approx \frac{1}{2}\hat{\mathbf{f}}^T\boldsymbol{K}^{-1}\hat{\mathbf{f}} - \log p(\boldsymbol{y}|\hat{\mathbf{f}}) + \frac{1}{2}\log|\boldsymbol{B}| \tag{5}$$

with $\boldsymbol{B} = \boldsymbol{I} + \boldsymbol{K}\boldsymbol{W}$. The first two terms are easily computed using the results of Newton's algorithm discussed in the previous paragraph, but the $\log|\boldsymbol{B}|$ and its gradients present difficulties. The $\boldsymbol{B}$ matrix is not a Kronecker product and a direct implementation of the log determinant has order $\mathcal{O}(N^3)$.

To reduce the computational complexity, we propose to approximate the prior covariance matrix

Algorithm 2 Laplace negative approximate marginal likelihood

function LAPLACEMARGLIK(covariance matrix $\boldsymbol{K}$, observations $\boldsymbol{y}$, likelihood function $p(\boldsymbol{y}|\boldsymbol{f})$)
  $[\hat{\mathbf{f}}, \boldsymbol{a}] \leftarrow$ Obtain from Algorithm 1
  for $d \leftarrow 1, \ldots, D$ **do**
    $\boldsymbol{Q}_d, \boldsymbol{S}_d \leftarrow \text{SCHUR}(\boldsymbol{K}_d)$ ▷ Eigen decomposition $\boldsymbol{K}_d = \boldsymbol{Q}_d \boldsymbol{S}_d \boldsymbol{Q}_d^T$
    $\boldsymbol{Q}_d, \boldsymbol{S}_d \leftarrow \text{REDUCERANK}(\boldsymbol{Q}_d, \boldsymbol{S}_d)$ ▷ Select largest eigenvalues
  end for
  $\boldsymbol{W} \leftarrow -\nabla\nabla \log p(\boldsymbol{y}|\boldsymbol{f})$
  $\boldsymbol{\Lambda}_1 \leftarrow \text{diag}(\text{diag}(\boldsymbol{K}) - (\boldsymbol{Q} \circ \boldsymbol{Q}) \cdot \text{diag}(\boldsymbol{S}))$ ▷ Eq. (6)
  $\boldsymbol{\Lambda}_2 \leftarrow \boldsymbol{I} + \boldsymbol{W}^{\frac{1}{2}} \boldsymbol{\Lambda}_1 \boldsymbol{W}^{\frac{1}{2}}$
  $\boldsymbol{\Lambda}_3 \leftarrow \boldsymbol{W}^{\frac{1}{2}} \boldsymbol{\Lambda}_2^{-1} \boldsymbol{W}^{\frac{1}{2}}$
  $\boldsymbol{L} \leftarrow \text{CHOLESKY}(\boldsymbol{S}^{-1} + \boldsymbol{Q}^T \boldsymbol{\Lambda}_3 \boldsymbol{Q})$ ▷ Eq. (9)
  $logdetB \leftarrow \sum_i \log \Lambda_{2ii} + \sum_i \log S_{ii} + 2 * \sum_i \log L_{ii}$
  return $-\log q(\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{\theta}) \leftarrow \frac{1}{2} \boldsymbol{a}^T \hat{\mathbf{f}} - \log p(\boldsymbol{y}|\hat{\mathbf{f}}) + \frac{1}{2} logdetB$ ▷ neg. approx. log. marg. lik.
end function

using a reduced-rank approximation:

$$\boldsymbol{K} \approx \boldsymbol{Q}\boldsymbol{S}\boldsymbol{Q}^T + \boldsymbol{\Lambda}_1, \text{ with } \boldsymbol{Q} = \bigotimes_{d=1}^{D} \boldsymbol{Q}_d, \boldsymbol{S} = \bigotimes_{d=1}^{D} \boldsymbol{S}_d, \boldsymbol{\Lambda}_1 = \text{diag}(\text{diag}(\boldsymbol{K}) - \text{diag}(\boldsymbol{Q}\boldsymbol{S}\boldsymbol{Q}^T)) \quad (6)$$

and where $\boldsymbol{K}$ is an $N \times N$ matrix, $\boldsymbol{Q}$ is an $N \times R$ matrix, and $\boldsymbol{S}$ is an $R \times R$ diagonal matrix with $R \ll N$. The $\boldsymbol{\Lambda}_1$ matrix is a diagonal matrix that preserves the exact full-rank diagonal to obtain a better approximation without raising the computational costs. The sum of a diagonal matrix and a low rank matrix allows for efficient computations using the Woodbury formula and matrix determinant lemma

$$(\boldsymbol{A} + \boldsymbol{U}\boldsymbol{W}\boldsymbol{V})^{-1} = \boldsymbol{A}^{-1} - \boldsymbol{A}^{-1}\boldsymbol{U}(\boldsymbol{W}^{-1} + \boldsymbol{V}\boldsymbol{A}^{-1}\boldsymbol{U})^{-1}\boldsymbol{V}\boldsymbol{A}^{-1} \quad (7)$$

$$|\boldsymbol{A} + \boldsymbol{U}\boldsymbol{W}\boldsymbol{V}^T| = |\boldsymbol{W}^{-1} + \boldsymbol{V}^T\boldsymbol{A}^{-1}\boldsymbol{U}||\boldsymbol{W}||\boldsymbol{A}| \quad (8)$$

and has been used widely before in statistics and machine learning [1, 7]. The *key insight* here is that the Kronecker product allows the decomposition $\boldsymbol{Q}\boldsymbol{S}\boldsymbol{Q}^T$ to be *efficiently* obtained in $\mathcal{O}(\sum_{d=1}^{D} N_d^3)$.

4.1 Evaluating the Marginal Likelihood

With the reduced-rank approximation the log determinant in the negative marginal likelihood can be evaluated more efficiently using the relation:

$$\begin{aligned} |\boldsymbol{B}| &= |\boldsymbol{I} + \boldsymbol{W}^{\frac{1}{2}}\boldsymbol{K}\boldsymbol{W}^{\frac{1}{2}}| \approx |\boldsymbol{I} + \boldsymbol{W}^{\frac{1}{2}}\boldsymbol{\Lambda}_1\boldsymbol{W}^{\frac{1}{2}} + \boldsymbol{W}^{\frac{1}{2}}\boldsymbol{Q}\boldsymbol{S}\boldsymbol{Q}^T\boldsymbol{W}^{\frac{1}{2}}| = |\boldsymbol{\Lambda}_2 + \boldsymbol{W}^{\frac{1}{2}}\boldsymbol{Q}\boldsymbol{S}\boldsymbol{Q}^T\boldsymbol{W}^{\frac{1}{2}}| \\ &= |\boldsymbol{\Lambda}_2||\boldsymbol{S}||\boldsymbol{S}^{-1} + \boldsymbol{Q}^T\boldsymbol{W}^{\frac{1}{2}}\boldsymbol{\Lambda}_2^{-1}\boldsymbol{W}^{\frac{1}{2}}\boldsymbol{Q}| = |\boldsymbol{\Lambda}_2||\boldsymbol{S}||\boldsymbol{S}^{-1} + \boldsymbol{Q}^T\boldsymbol{\Lambda}_3\boldsymbol{Q}| \end{aligned} \quad (9)$$

with $\boldsymbol{\Lambda}_2 = \boldsymbol{I} + \boldsymbol{W}^{\frac{1}{2}}\boldsymbol{\Lambda}_1\boldsymbol{W}^{\frac{1}{2}}$ and $\boldsymbol{\Lambda}_3 = \boldsymbol{W}^{\frac{1}{2}}\boldsymbol{\Lambda}_2^{-1}\boldsymbol{W}^{\frac{1}{2}}$. The final formula can be evaluated efficiently since it consists of two diagonal matrices and one low-rank matrix. To evaluate the last term we use the Cholesky decomposition $\boldsymbol{L} = \text{CHOLESKY}(\boldsymbol{S}^{-1} + \boldsymbol{Q}^T\boldsymbol{\Lambda}_3\boldsymbol{Q})$, which can be reused in the gradients. Pseudocode is given in Algorithm 2.

4.2 Gradients

We can optimize the marginal likelihood with respect to the hyperparameters $\boldsymbol{\theta}$. For this we need the partial derivatives $\partial q(\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{\theta})/\theta_j$. Since $\hat{\mathbf{f}}$ and $\boldsymbol{W}$ are implicit functions of $\boldsymbol{\theta}$ we need to take care of both the explicit and implicit derivatives [8].

$$\frac{\partial \log q(\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{\theta})}{\partial \theta_j} = \left.\frac{\partial \log q(\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{\theta})}{\partial \theta_j}\right|_{\text{explicit}} + \sum_{i=1}^{N} \frac{\partial \log q(\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{\theta})}{\partial \hat{f}_i} \frac{\partial \hat{f}_i}{\partial \theta_j} \quad (10)$$

4.2.1 Explicit Derivatives – Kernel Hyperparameters

The explicit derivatives are given by:

$$\left.\frac{\partial \log q(\boldsymbol{y}|\boldsymbol{X},\boldsymbol{\theta})}{\partial \theta_j^c}\right|_{\text{explicit}} = \frac{1}{2}\hat{\mathbf{f}}^T \boldsymbol{K}^{-1}\frac{\partial \boldsymbol{K}}{\partial \theta_j^c}\boldsymbol{K}^{-1}\hat{\mathbf{f}} - \frac{1}{2}\text{tr}\left((\boldsymbol{W}^{-1}+\boldsymbol{K})^{-1}\frac{\partial \boldsymbol{K}}{\partial \theta_j^c}\right). \tag{11}$$

The first term is easily computed using Kronecker matrix-vector products, but we approximate the second term using the low-rank decomposition. Since

$$\begin{aligned}(\boldsymbol{W}^{-1}+\boldsymbol{K})^{-1} &= \boldsymbol{W}^{\frac{1}{2}}(\boldsymbol{I}+\boldsymbol{W}^{\frac{1}{2}}\boldsymbol{K}\boldsymbol{W}^{\frac{1}{2}})^{-1}\boldsymbol{W}^{\frac{1}{2}} \\ &\approx \boldsymbol{W}^{\frac{1}{2}}(\boldsymbol{I}+\boldsymbol{W}^{\frac{1}{2}}\boldsymbol{\Lambda}_1\boldsymbol{W}^{\frac{1}{2}}+\boldsymbol{W}^{\frac{1}{2}}\boldsymbol{Q}\boldsymbol{S}\boldsymbol{Q}^T\boldsymbol{W}^{\frac{1}{2}})^{-1}\boldsymbol{W}^{\frac{1}{2}} \\ &= \boldsymbol{W}^{\frac{1}{2}}(\boldsymbol{\Lambda}_2+\boldsymbol{W}^{\frac{1}{2}}\boldsymbol{Q}\boldsymbol{S}\boldsymbol{Q}^T\boldsymbol{W}^{\frac{1}{2}})^{-1}\boldsymbol{W}^{\frac{1}{2}} \\ &= \boldsymbol{W}^{\frac{1}{2}}(\boldsymbol{\Lambda}_2^{-1}-\boldsymbol{\Lambda}_2^{-1}\boldsymbol{W}^{\frac{1}{2}}\boldsymbol{Q}(\boldsymbol{S}^{-1}+\boldsymbol{Q}^T\boldsymbol{\Lambda}_3^{-1}\boldsymbol{Q})^{-1}\boldsymbol{Q}^T\boldsymbol{W}^{\frac{1}{2}}\boldsymbol{\Lambda}_2^{-1})\boldsymbol{W}^{\frac{1}{2}} \\ &= \boldsymbol{\Lambda}_3-\boldsymbol{\Lambda}_3\boldsymbol{Q}(\boldsymbol{S}^{-1}+\boldsymbol{Q}^T\boldsymbol{\Lambda}_3\boldsymbol{Q})^{-1}\boldsymbol{Q}^T\boldsymbol{\Lambda}_3\end{aligned} \tag{12}$$

we can rewrite the trace term as follows

$$\begin{aligned}\text{tr}\left(\left(\boldsymbol{W}^{-1}+\boldsymbol{K}\right)^{-1}\frac{\partial \boldsymbol{K}}{\partial \theta_j^c}\right) &= \text{tr}\left(\boldsymbol{\Lambda}_3\frac{\partial \boldsymbol{K}}{\partial \theta_j^c}\right) - \text{tr}\left(\boldsymbol{\Lambda}_3\boldsymbol{Q}\left(\boldsymbol{S}^{-1}+\boldsymbol{Q}^T\boldsymbol{\Lambda}_3\boldsymbol{Q}\right)^{-1}\boldsymbol{Q}^T\boldsymbol{\Lambda}_3\frac{\partial \boldsymbol{K}}{\partial \theta_j^c}\right) \\ &= \text{tr}\left(\boldsymbol{\Lambda}_3\frac{\partial \boldsymbol{K}}{\partial \theta_j^c}\right) - \text{tr}\left(\boldsymbol{V}_1^T\frac{\partial \boldsymbol{K}}{\partial \theta_j^c}\boldsymbol{V}_1\right)\end{aligned} \tag{13}$$

with $\boldsymbol{V}_1 = \boldsymbol{\Lambda}_3\boldsymbol{Q}\text{CHOLESKY}\left[(\boldsymbol{S}^{-1}+\boldsymbol{Q}^T\boldsymbol{\Lambda}_3\boldsymbol{Q})^{-1}\right]$. In this last line we also used the cyclic property of the trace operator $\text{tr}(\boldsymbol{ABC}) = \text{tr}(\boldsymbol{BCA})$ to obtain a trace of a low-rank $R \times R$ matrix.

4.2.2 Implicit Derivatives – Kernel Hyperparameters

To evaluate the implicit derivatives we need the following expression:

$$\frac{\partial \log q(\boldsymbol{y}|\boldsymbol{X},\boldsymbol{\theta})}{\partial \hat{f}_i} = -\frac{1}{2}\left[(\boldsymbol{K}^{-1}+\boldsymbol{W})^{-1}\right]_{ii}\frac{\partial^3}{\partial f_i^3}\log p(\boldsymbol{y}|\hat{\mathbf{f}}) \tag{14}$$

The troublesome part can be rewritten using the low-rank decomposition as follows

$$\begin{aligned}(\boldsymbol{K}^{-1}+\boldsymbol{W})^{-1} &= (\boldsymbol{K}^{-1}+\boldsymbol{W}^{\frac{1}{2}}\boldsymbol{I}\boldsymbol{W}^{\frac{1}{2}})^{-1} \\ &= \boldsymbol{K}-\boldsymbol{K}\boldsymbol{W}^{\frac{1}{2}}(\boldsymbol{I}+\boldsymbol{W}^{\frac{1}{2}}\boldsymbol{K}\boldsymbol{W}^{\frac{1}{2}})^{-1}\boldsymbol{W}^{\frac{1}{2}}\boldsymbol{K} \\ &= \boldsymbol{K}-\boldsymbol{K}(\boldsymbol{W}^{-1}+\boldsymbol{K})^{-1}\boldsymbol{K} \\ &\approx \boldsymbol{K}-\boldsymbol{K}\boldsymbol{\Lambda}_3\boldsymbol{K}+\boldsymbol{K}\boldsymbol{\Lambda}_3\boldsymbol{Q}(\boldsymbol{S}^{-1}+\boldsymbol{Q}^T\boldsymbol{\Lambda}_3\boldsymbol{Q})^{-1}\boldsymbol{Q}^T\boldsymbol{\Lambda}_3\boldsymbol{K} \\ &= \boldsymbol{K}-\boldsymbol{K}\boldsymbol{\Lambda}_3\boldsymbol{K}+\boldsymbol{V}_2\boldsymbol{V}_2^T\end{aligned} \tag{15}$$

with $\boldsymbol{V}_2 = \boldsymbol{K}\boldsymbol{\Lambda}_3\boldsymbol{Q}\text{CHOLESKY}\left[(\boldsymbol{S}^{-1}+\boldsymbol{Q}^T\boldsymbol{\Lambda}_3\boldsymbol{Q})^{-1}\right] = \boldsymbol{K}\boldsymbol{V}_1$. The diagonal $\text{diag}[(\boldsymbol{K}^{-1}+\boldsymbol{W})^{-1}]$ can be easily obtained using Eq. (15). The remaining term

$$\frac{\partial \hat{\mathbf{f}}}{\partial \theta_j^c} = (\boldsymbol{I}+\boldsymbol{K}\boldsymbol{W})^{-1}\frac{\partial \boldsymbol{K}}{\partial \theta_j^c}\nabla \log p(\boldsymbol{y}|\hat{\mathbf{f}}) \tag{16}$$

can be solved using conjugate gradient avoiding the need to directly compute the expensive matrix inverse.

4.2.3 Explicit Derivatives – Likelihood Hyperparameters

The explicit derivatives of the likelihood are given by:

$$\left.\frac{\partial \log q(\boldsymbol{y}|\boldsymbol{X},\boldsymbol{\theta})}{\partial \theta_j^l}\right|_{\text{explicit}} = \frac{\partial}{\partial \theta_j^l}\log p(\boldsymbol{y}|\boldsymbol{f}) + \frac{1}{2}\left[(\boldsymbol{K}^{-1}+\boldsymbol{W})^{-1}\right]_{ii}\frac{\partial}{\partial \theta_j^l}\nabla^2 \log p(\boldsymbol{y}|\boldsymbol{f}) \tag{17}$$

in which the second term can be approximated as outlined in Eq. (15).

Algorithm 3 Laplace gradients

1: **function** LAPLACEGRADIENT(covariance matrix $\boldsymbol{K}$, observations $\boldsymbol{y}$, likelihood function $p(\boldsymbol{y}|\boldsymbol{f})$)
2: $[\hat{\mathbf{f}}, \boldsymbol{Q}, \boldsymbol{L}, \boldsymbol{\Lambda}_3] \leftarrow$ Obtain from Algorithm 1 and Algorithm 2
3: $\boldsymbol{V}_1 \leftarrow \boldsymbol{\Lambda}_3\boldsymbol{Q}/\boldsymbol{L}$ ▷ $\boldsymbol{V}_1 = \boldsymbol{\Lambda}_3\boldsymbol{Q}\text{CHOLESKY}\left[(\boldsymbol{S}^{-1} + \boldsymbol{Q}^T\boldsymbol{\Lambda}_3\boldsymbol{Q})^{-1}\right]$
4: $\boldsymbol{V}_2 \leftarrow \boldsymbol{K}\boldsymbol{V}_1$
5: $\boldsymbol{C} \leftarrow \text{diag}(\boldsymbol{K}) - (\boldsymbol{K} \circ \boldsymbol{K}) \cdot \text{diag}(\boldsymbol{\Lambda}_3) + \text{SUMROWS}(\boldsymbol{V}_2 \circ \boldsymbol{V}_2)$ ▷ Eq. (15)
6: $s_2 \leftarrow \frac{1}{2}\boldsymbol{C} \cdot \nabla^3 \log p(\boldsymbol{y}|\boldsymbol{f})$ ▷ Eq. (14)
7: **for** $j \leftarrow 1, \ldots, \dim(\boldsymbol{\theta}^c)$ **do** ▷ loop over kernel hyperparameters
8: $\boldsymbol{s}_1 \leftarrow \frac{1}{2}\boldsymbol{a}^T \frac{\partial \boldsymbol{K}}{\partial \theta_j^c}\boldsymbol{a} - \frac{1}{2}\text{diag}(\boldsymbol{\Lambda}_3)^T \text{diag}(\frac{\partial \boldsymbol{K}}{\partial \theta_j^c}) + \frac{1}{2}\text{tr}(\boldsymbol{V}_1^T \frac{\partial \boldsymbol{K}}{\partial \theta_j^c}\boldsymbol{V}_1)$ ▷ Eqs. (11–13)
9: Iteratively solve: $(\boldsymbol{I} + \boldsymbol{K}\boldsymbol{W})\boldsymbol{s}_3 = \boldsymbol{K}\nabla \log p(\boldsymbol{y}|\boldsymbol{f})$ ▷ Eq. (16) using conjugate gradient
10: $\nabla_j - \log p(\boldsymbol{y}|\boldsymbol{f}) \leftarrow -\boldsymbol{s}_1 - \boldsymbol{s}_2^T\boldsymbol{s}_3$ ▷ Gradient $-\frac{\partial}{\partial \theta_j^c}\log p(\boldsymbol{y}|\boldsymbol{f})$
11: **end for**
12: **for** $j \leftarrow 1, \ldots, \dim(\boldsymbol{\theta}^l)$ **do** ▷ loop over likelihood hyperparameters
13: $\boldsymbol{s}_1 \leftarrow \frac{\partial}{\partial \theta_j^l}\log p(\boldsymbol{y}|\boldsymbol{f}) + \frac{1}{2}\boldsymbol{C}\frac{\partial}{\partial \theta_j^l}\nabla^2 \log p(\boldsymbol{y}|\boldsymbol{f})$ ▷ Eq. (17)
14: Iteratively solve: $(\boldsymbol{I} + \boldsymbol{K}\boldsymbol{W})\boldsymbol{s}_3 = \boldsymbol{K}\frac{\partial}{\partial \theta_j^l}\nabla \log p(\boldsymbol{y}|\boldsymbol{f})$ ▷ Eq. (18) using conjugate gradient
15: $-\nabla_j \log p(\boldsymbol{y}|\boldsymbol{f}) \leftarrow -\boldsymbol{s}_1 - \boldsymbol{s}_2^T\boldsymbol{s}_3$ ▷ Gradient $-\frac{\partial}{\partial \theta_j^l}\log p(\boldsymbol{y}|\boldsymbol{f})$
16: **end for**
17: **return** $-\nabla \log p(\boldsymbol{y}|\boldsymbol{f})$ ▷ Gradient $-\frac{\partial}{\partial \boldsymbol{\theta}}\log p(\boldsymbol{y}|\boldsymbol{f})$
18: **end function**

4.2.4 Implicit Derivatives – Likelihood Hyperparameters

The implicit derivatives of the likelihood (cf. Eq. (10)) are computed by combining Eq. (14) with

$$\frac{\partial \hat{\mathbf{f}}}{\partial \theta_j^l} = (\boldsymbol{I} + \boldsymbol{K}\boldsymbol{W})^{-1}\boldsymbol{K}\frac{\partial}{\partial \theta_j^l}\nabla \log p(\boldsymbol{y}|\boldsymbol{f}) \tag{18}$$

of which the latter is solved using conjugate gradient.

5 Memory and Run Time Complexity of the Algorithms

Algorithms 1, 2, and 3 are both *memory* and *run time* efficient. Care is taken to never evaluate a full $N \times N$ matrix, which may not fit into memory. The matrices $\boldsymbol{W}, \boldsymbol{b}, \boldsymbol{\Lambda}_1, \boldsymbol{\Lambda}_2, \boldsymbol{\Lambda}_3$ are diagonal each requiring $\mathcal{O}(N)$ storage, $\boldsymbol{K}$ is a Kronecker matrix requiring $\mathcal{O}(\sum_{d=1}^{D} N_d^2)$ storage, $\boldsymbol{S} = \otimes \boldsymbol{S}_d$ requires at most $\mathcal{O}(R)$ storage, $\boldsymbol{L}$ requires $\mathcal{O}(R^2)$ storage, and $\boldsymbol{Q} = \otimes \boldsymbol{Q}_d, \boldsymbol{V}_1, \boldsymbol{V}_2$ require (at most) $\mathcal{O}(RN)$ storage. Because all matrix and vector operations operate on small, i.e., $R \times N$, diagonal, or Kronecker matrices, instead of full $N \times N$ matrices, an increase in efficiency is obtained when N becomes large. Note that in Algorithm 2, line 8 and Algorithm 3, line 5 we made use of the identity

$$\text{diag}(\boldsymbol{A}\boldsymbol{B}\boldsymbol{C}^T) = (\boldsymbol{A} \circ \boldsymbol{C}) \cdot \text{diag}(\boldsymbol{B}) \tag{19}$$

which holds when $\boldsymbol{B}$ is a diagonal matrix to speed up computations. For a Kronecker matrix $\boldsymbol{K}$, the Hadamard product $\boldsymbol{K} \circ \boldsymbol{K}$ can be evaluated efficiently and is again a Kronecker matrix.

6 Experiments

In order to compare the run time complexity of a standard Gaussian process with a Gaussian process having a Kronecker product kernel we implemented the algorithms in the publicly available Gaussian process toolbox GPstuff [12] and ran the algorithms on a synthetic dataset. As input we use gridded data on the D-dimensional hypercube $\mathcal{X} = \prod_{d=1}^{D}[0, 1]$. We consider the binary classification task of a datapoint belonging to the D-dimensional unit ball in $\mathcal{X}$.

For the likelihood model we use a probit model $p(y_i|f_i) = \Phi(y_i f_i)$ with $y_i \in \{-1, 1\}$. We generated inputs with $D = 2$ using varying grid sizes and ran both the standard Laplace approximation and the

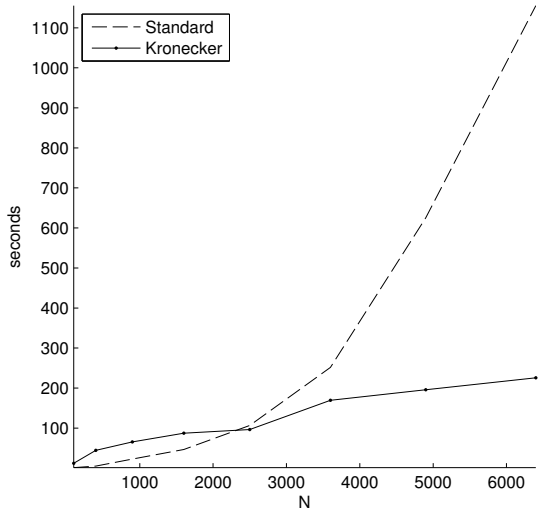


Figure 1: Run time of the standard Laplace algorithm versus the Kronecker Laplace algorithm on a 2D classification problem with a varying number of input data points.

Kronecker Laplace approximation and measured their run-time averaged over five runs. The results are shown in Figure 1. For the low-rank approximation we only selected the eigenvalues higher than 1E-6, but if necessary only select from the highest eigen values such that the rank is below 10% of the number of input data points. The tolerance in the conjugate gradient iterations was set to 1E-10. Clearly, the increase in computational complexity with increasing data size is much lower for the Kronecker Laplace approximation than for the standard Laplace approximation.

7 Conclusions

In this paper we have presented a scalable Gaussian process framework for gridded inputs and non-Gaussian observation models that factorize over cases. We used the Laplace approximation to perform approximate inference and showed it can efficiently be combined with tensor product kernels using fast matrix-vector operations and an efficiently obtained low-rank approximation of the kernel matrix. We provided derivations and pseudocode of the algorithms and empirically validated our approach on a binary classification problem showing a major performance improvement in terms of run time.

Directions for further work include among others handling missing observations and a more detailed analysis of the computational complexity. The standard Laplace approximation is quite straightforward, but the Kronecker Laplace approximation provides more options that each influence the computational complexity and quality of the algorithms. These include, among others, the rank of the low-rank approximation, the tolerances used in the Conjugate Gradients Iterated solvers, and the number of hyper-parameters.

References

[1] N. Cressie and G. Johannesson. Fixed rank kriging for very large spatial data sets. *Journal of the Royal Statistical Society*, 70:209–226, 2008.

[2] J. P. Cunningham, K. V. Shenoy, and M. Sahani. Fast Gaussian process methods for point process intensity estimation. In *Proceedings of the 25th international conference on Machine learning*, pages 192–199, 2008.

[3] E. Gilboa, Y. Saatci, and J. P. Cunningham. Scaling multidimensional Gaussian processes using projected additive approximations. In *International Conference on Machine Learning*, 2013.

[4] A. Knutson and T. Tao. Honeycombs and sums of Hermitian matrices. *Notices Amer. Math. Soc*, pages 175–186, 2000.

[5] C. F. van Loan. The ubiquitous Kronecker product. *Journal of Computational and Applied Mathematics*, 123(1–2):85–100, 2000.

[6] J. Quiñonero-Candela and C. E. Rasmussen. A unifying view of sparse approximate Gaussian process regression. *Journal of Machine Learning Research*, 6:1939–1959, December 2005.

[7] J. Quiñonero-Candela, C.E. Rasmussen, and C.K.I. Williams. Approximation methods for Gaussian process regression. *Advances in Neural Information Processing Systems*, pages 203–223, 2007.

[8] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

[9] J. Riihimäki and A. Vehtari. Laplace approximation for logistic Gaussian process density estimation. *Bayesian Analysis*, in press, 2014.

[10] E. Snelson and Z. Ghahramani. Sparse Gaussian processes using pseudo-inputs. *Advances in Neural Information Processing Systems*, 2006.

[11] O. Stegle, C. Lippert, J. M. Mooij, N. D. Lawrence, and K. Borgwardt. Efficient inference in matrix-variate Gaussian models with iid observation noise. In *Advances in Neural Information Processing Systems 24*, pages 630–638, 2011.

[12] J. Vanhatalo, J. Riihimäki, J. Hartikainen, P. Jylänki, V. Tolvanen, and A. Vehtari. GPstuff: Bayesian modeling with Gaussian processes. *Journal of Machine Learning Research*, 14(Apr):1175–1179, 2013.

[13] C. K. I. Williams and D. Barber. Bayesian classification with Gaussian processes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(12):1342–1351, 1998.

[14] A. G. Wilson, E. Gilboa, A. Nehorai, and J. P. Cunningham. GPatt: Fast multidimensional pattern extrapolation with Gaussian processes. *arXiv, 2013.*

Predicting Pseudo-Random Behaviour in Professional Sports

Manuel Kauschinger Kurt Driessens

Department of Knowledge Engineering, Maastricht University, The Netherlands

Abstract

Truly random behaviour is difficult for humans to generate, and they often unconsciously fall back to repetitive sequences that only appear to be random. In sports, discovering these regularities could yield a substantial benefit, as athletes often try to exhibit random behaviour in order to prevent their opponents from foreseeing the planned strategy. This paper presents a case study where statistical and machine learning methods are employed in order to predict the serve placement of professional tennis players. Surprisingly, results show that professional tennis players manage to mix up their actions such that it is difficult to accurately predict the location of the next serve.

1 Introduction

In many sports, players are advised to mix up their strategies in order to be unpredictable for the opponent. Given some reasonable assumptions, such as the difficulty and rate of success being equal for all alternatives, this would mean that players are better off if they choose their actions such that every possible move is equally likely. In tennis the unpredictability becomes an important strategy when serving the ball. If the player is not able to mix up the serve strategy, the opponent can use this to his advantage by positioning himself in a way to increase his chances to score a point. Since professional tennis players are often comparably strong, predictability could be a decisive point. However, psychological literature suggests that humans are bad in coming up with random sequences and thus can be analysed and predicted [5]. Thus, it might be interesting for trainers and players to analyse and predict the behaviour of the opponent in order to obtain an advantage. The question arises whether tennis players are able to play in an unpredictable way or is it possible to build appropriate predictors for the players behaviour.

In 2004, Daryl Fisher wrote an article in which he suggests a few techniques for professional tennis players, to act such that the opponent has no chance to predict the next placement of a serve. One of his ideas was to use random-number generators as a source for unpredictable serves. In the course of his article he also proposes a stronger technique to get random numbers, that is to use the last digit from a heart rate monitor in order to decide the next location he is going to serve [2]. We have little insight into whether these techniques are actually used or not, but they do illustrate the importance of the randomisation of actions.

The objective of our case study is to attempt to build a model that predicts the location of the next serve and analyse the randomness of professional tennis players using techniques from machine learning and statistical analysis. Considering the restrictions imposed by the possible application of the techniques, the approaches should allow online learning and prediction from a relatively short history of serves. In the following sections, we formally define the prediction task and discuss the mathematical models employed. Furthermore, the results obtained are presented, analysed and discussed.

2 Task Definition

The task of predicting the next serve location in tennis dependent on previous serving history, can be formulated as follows:

Given: A sequence of events $X = \{x(0), \ldots, x(t)\}$ encoded as a string of symbols over an alphabet Σ, where t indicates discrete time.

Task: Make a prediction $x(t+1)$ based on a substring of length $n+1$: $X' = \{x(t-n), \ldots, x(t)\}$.

To measure the accuracy of the predictions a sequence of length $n+1$ is used for training and predicting the next symbol to appear. An average accuracy is computed over the whole sequence as $\frac{\sum_{i=1}^{t} \delta(x(i),p(i))}{t}$ where $p(i)$ is the prediction made at time i and $\delta(x,y) = 1$ in case $x = y$ and 0 otherwise. The success rate of the classifier can be determined by repeating this technique for different lengths n of the training data.

3 Tested Approaches

A number of mathematical models for the task described in Section 2 exist. Three were selected and are described below.

3.1 Markov Chains

A Markov chain is a stochastic process that changes its state over time [6]. Given a set of states $S = \{s_1, s_2, \ldots, s_m\}$, the process moves from one state to another (or itself) at each discrete time step. Each move is called a step. A Markov chain is fully defined by the transition probabilities $p_{i,j}$ that represent that given a chain currently in state s_i, it moves with probability $p_{i,j}$ to the state s_j in the next time step, i.e., $p_{i,j} = Pr(s_j|s_i)$. This means that learning a Markov chain requires the estimation of m^2 model parameters where m represents the number of states.

To employ a Markov chain to the serve-prediction problem, we could model each serve location (e.g. "left", "middle" and "right") as a state of the chain, and try to predict which state/location the chain will move to next. However, a standard or first-order Markov chain is memory-less, which means, the probability of going from state s_1 to state s_2 depends only on s_1 and s_2, and not on how state s_1 was reached, or, in the setting as defined above $P(x(t+1)|x(t-n), \ldots, x(t)) = P(x(t+1)|x(t))$. This means that the first order Markov Chain computes the conditional probabilities by considering only the last symbol in the string, i.e., the single previous serve only.

To take more history into account an n-order Markov model was used. The algorithm takes the last n symbols in the string to compute the conditional probabilities, in fact treating every possible string of n symbols as a separate state. However, setting $n > 1$ also raises the number of model parameters to be estimated. In fact, an n-order Markov chain using an alphabet Σ will require $(m^n)^2$ parameters to be trained, where $m = \#\Sigma$ is the size of the vocabulary. Given the limited length of observable sequences in tennis serve prediction, the order of the used Markov chain will have to be severely limited.

3.2 Prediction Suffix Tree

The Prediction Suffix Tree (PST) [3] is often used in bio-informatics and provides excellent results for the modelling of protein families [1]. The PST is an effective and well studied depth limited tree structure that stores all sub-sequences of a given sequence of symbols and can be used to make a prediction about the next symbol based on the symbols observed so far. For example, if we consider a suffix tree with depth $d = 2$ over the sequence $X = (LLRLLRLLR)$ then the PST is described in Figure 1. The empirical probabilities for each symbol are computed over the whole sequence. Prediction Suffix Trees can predict the next symbol in the sequence by scanning the tree for different suffix lengths and predict the symbol with the highest conditional probability. Since the statistical correlation can be expected to decrease as the time between events increases [9] the contribution of each suffix is multiplied by a weight factor w that decreases with the length of the suffix:

$$w = \frac{1}{d}$$

PSTs have the advantage over other Markov models that they automatically prune branches where the conditional probabilities are not defined. This reduces storage costs substantially. In fact, the symbol

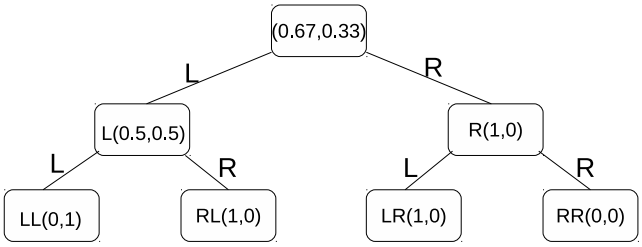


Figure 1: Prediction Suffix Tree for the sequence $X = (LLRLLRLLR)$. The string is decomposed into suffixes, where each node in the tree shows the conditional probabilities for the next letter in the string. The first number is the conditional probability that the next symbol will be a L and the second number is the probability that the next symbol is a R. For example, the last symbol in the string is a L, then the probability that the next symbol will be a L is 0.5. Likewise if the last two symbols in the string are R and L the probability that the next symbol will be a L is 1.

probability computed by a PST can be compared to a weighted joint probability from all n-order Markov chains ($n < d$) for which a fitting observation was made.

3.3 Artificial Neural Network

An Artificial Neural Network is composed of a number of highly interconnected processing elements (logistic regression nodes) working together to solve non-linear prediction problems. Neural networks learn from training examples by comparing the output of the network with the desired output of the training example and updating the weights of the connections in the inner structure. More detailed information on how such models are commonly implemented can be found in [8]. For the training of Artificial Neural Networks on the available data several algorithms can be used. The most popular is the back propagation algorithm which is described in [4].

In order to apply the neural network to time series prediction, a sliding window approach was used. It uses the last n observations as inputs to the network and a single output that represents the next observation. The n-sized window slides over the full training sequence in order to learn the network, possibly multiple times to give the weights of the network time to converge. Figure 2 gives an illustration of the basic architecture.

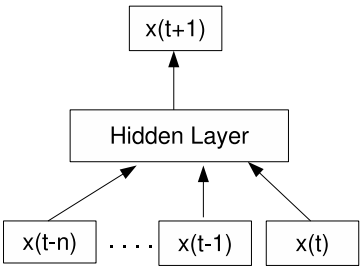


Figure 2: Sliding Window Neural Network for time series prediction with history length n.

In this set-up, the number of parameters to train is $n_i \times n_h + n_h$ where n_i is the window length and n_h the number of hidden nodes used. Again, keeping in mind the limited amount of training data available, the number of nodes will have to be kept very small.

4 Initial Results

For the case study on tennis-serve-prediction, we divided the service box into 3 regions: “left”, “middle” and “right”. We gathered data about the shot placement tendencies for two Dutch professional tennis players from 3 matches during the tennis season in the years 2012 and 2013. The location for each serve

in the opponents court is represented as a symbol over the alphabet $\Sigma = \{L, M, R\}$. All serves from the chosen tennis player are stored as a sequence of characters using the alphabet Σ.

Because of the speed and drill a player has to put into the first serve, chances are higher that the serve goes out of the service box. The second serve is usually more conservative in order to avoid a double fault. Therefore we have also recorded the desired location of the first serve and the location of the second serve, in the hope to increase the accuracy of the classifier. The information of the first and the second serve are in that case stored in separate sequences. If the classifier detects a fault in the sequence of first serves it can then switch to a second classifier trained on the sequence of second serves.

As stated before, it is hard for humans to behave randomly. We expect that even with the limited amount of training data available, the models should be able to predict serve-locations quite accurately.

4.1 Thiemo De Bakker

Thiemo De Bakker is a 25 years old professional tennis player, born in Den Haag Netherlands. He obtained his highest Association of Tennis Professionals (ATP) rating of 40 in 2010. Currently he is ranked number 134. He is a very successful tennis player who has won a lot of prestigious titles since the beginning of his career. We logged his serving locations from a tennis game in 2013.

To give the models some time to train, we used an initial training sequence of 25 symbols, meaning we do not require the system to make predictions on the first 25 serves.[1] The recorded sequence for this tennis match consists of 70 serves. We tested both a first and second order Markov chain, neural networks with a window length 3 and both 3 and 6 hidden nodes and a PST with maximum depth 5. Table 1 shows the results of this experiment.

Algorithm	Correct predictions
1^{st} order Markov Chain	36.36%
2^{nd} order Markov Chain	29.54%
Neural Network (3,3,1)	38.63%
Neural Network (3,6,1)	37.36%
PST	40.90%

Table 1: Predictive performance for Thiemo de Bakker

Algorithm	Correct predictions
1^{st} order Markov Chain	36.55%
2^{nd} order Markov Chain	31.33%
Neural Network (3,3,1)	39.67%
Neural Network (3,6,1)	38.36%
PST	42.43%

Table 2: Predictive performance with separate second serve

We observe a best performance of 40.90% using the PST. The neural network with 3 input and 3 hidden nodes performs with 38.63% correct predictions second best on the data. For the two Markov Chain implementations we observe a performance close to random guessing, being 33.33. In fact, these results seem to indicate that Thiemo manages to distribute his serves so that they are difficult to anticipate.

In order to enhance the prediction accuracy of the classifiers we also take the records about missing a shot into account. The data set consists of 51 first serves and 19 faults (second serves). We split the sequences into two parts, one for first serves and the other whenever the first serve is a fault. If the player misses one shot we also record the desired location of the first serve in order to find a relation between the first and the second attempt. The transition probabilities for the first and second attempt are computed with the first order Markov chain since this is a good method to find relations between two serve locations. The result of treating the two sequences separately are shown in Table 2. We obtain only a slight increase in accuracy for all classifiers. The PST obtained the best results with 42.43% correct predictions while the two Markov models again perform close to a random classifier.

4.2 Michaella Krajicek

Michaella Krajicek was born in 1989. She is tennis professional since 2003 and has won three Women's Tennis Association (WTA) matches in her career. Michaella obtained her highest WTA rating in 2006 and is currently ranked 74.

Data was recorded for a match containing 100 serves. Applying the classifiers to Michaella's serve sequence as before we obtained the results for each classifier, shown in Table 3. Probably due to the

[1] We realise this is quite a substantial number, but it only confirms our conclusions. The exact choice for 25 is explained later in the text.

Algorithm	Correct predictions
1^{st} order Markov Chain	33.33%
2^{nd} order Markov Chain	36.00%
Neural Network (3,3,1)	45.33%
Neural Network (3,6,1)	41.33%
PST	39.66%

Table 3: Predictive performance for Michaella Krajicek

Algorithm	Correct predictions
1^{st} order Markov Chain	33.33%
2^{nd} order Markov Chain	34.66%
Neural Network (3,3,1)	46.00%
Neural Network (3,6,1)	41.63%
PST	40.25%

Table 4: Predictive performance with separate second serve sequence

higher number of serves in this match, the neural network shows with 45.33% correct predictions the best performance. The PST with 39.66% correct predictions is the second best classifier. Again, the two Markov models again perform close to a random classifier.

Again we attempt to improve the accuracy by treating the first and the second serve separately as explained above. From a total of 100 serves in this sequence, Michaella misses only 15 serves. Table 4 shows the results for this experiment. The two Markov Chains again perform close to a random classifier while the PST and the two neural networks slightly increase in accuracy. This seems to confirm that tennis players are more predictable on their second serve. However, reached accuracies are still disappointingly low, especially regarding the 25 first serves leeway given to the models.

5 Analysing "Normal" Human Random Behaviour

Since the results do not conform to our expectations, we decided to test and possibly tune the three listed approaches on more easily obtainable data. We logged 10 pseudo-random sequences of length 100 generated by several (untrained) volunteers that were asked to select randomly from three possible symbols. The different models were trained on the prefix and asked to predict the next value for the entire observed sequence, i.e., this time without the lead-in of 25 symbols.

Figure 3 (left) shows the learning rate for different length of training data when applying the first order Markov model. The x-axis labels the size of the used training data (i.e., the size of the prefix) while the y-axis shows the prediction accuracy obtained on the classifier. Naturally, this curve shows an increasing performance as the size of the prefix grows. The accuracy levels of at approximately 64% after an initial training sequence of 15 symbols.

The second order Markov Chain in comparison performs with 77% accuracy much better on the test data. However, the accuracy is very unstable for small training sequences as can be seen from Figure 3 (right) only reaching a reasonably stable performance after 23 symbols. This behaviour can be expected because the second order Markov model has 9 states and therefore 81 transition probabilities to estimate.

In order to find a suitable maximum depth for the PST, a number of experiments were conducted first that used an initial training sequence with a length of 30 symbols. The table on the left of Figure 4 shows the performance of the PST with different maximum history length. Based on the results of this experiment we chosen a suffix length of 5 symbols because it is the lowest value that shares the best error rate on the test data.

Applying the algorithm to the test data in the same manner as before, i.e., without the lead-in ex-

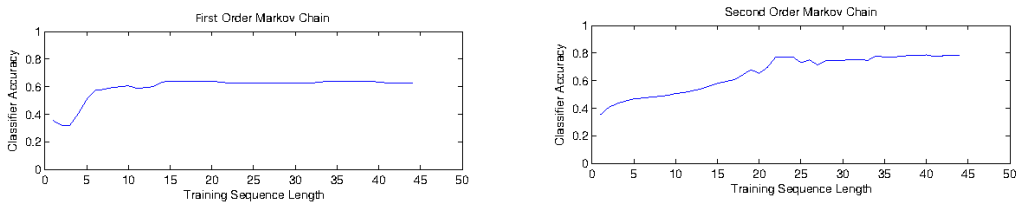


Figure 3: Prediction accuracy plot for the first order (left) and second order (right) Markov Chain

Suffix length	error
3	0.24
4	0.23
5	0.21
6	0.21
7	0.21

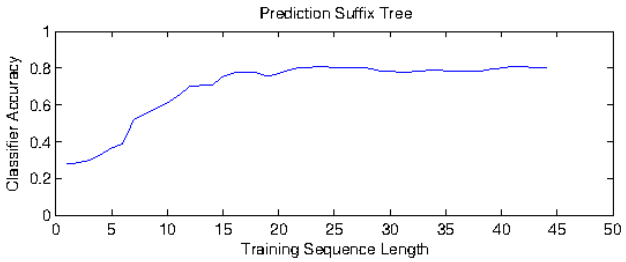


Figure 4: Prediction Suffix Tree performance. Left: Error rates for different max-depths. Right: PST learning rate for different lengths training sequences

amples, it can be observed in Figure 4 (right) that the PST manages to predict with an accuracy of 79% using 15 symbols for training. This indicates that the Prediction Suffix Tree should be a good method to address random behaviour prediction, since the classifier can make accurate predictions even for a short training sequence of 15 symbols.

Again, since it is difficult to determine the best number of input and hidden units in a neural network, we tested several networks configurations. Each neural network was trained with a training sequence of 40 symbols. The training was continued until the change of the error was below 1% for a period of 1000 iterations over the training set. The table on the left of Figure 5 shows the performance of such networks on the test data. The first column specifies the structure of the input layers, while the second column specifies the structure of the hidden layers. The last column shows the error rates obtained over the validation set. Based on the results, we have chose the two best networks with 3 input nodes and 3 or 6 hidden nodes for our model.

Input layer	Hidden layer	error
3	3	0.26
6	3	0.28
10	3	0.46
3	6	0.26
6	6	0.29
10	6	0.47
3	9	0.28
6	9	0.28
10	9	0.45

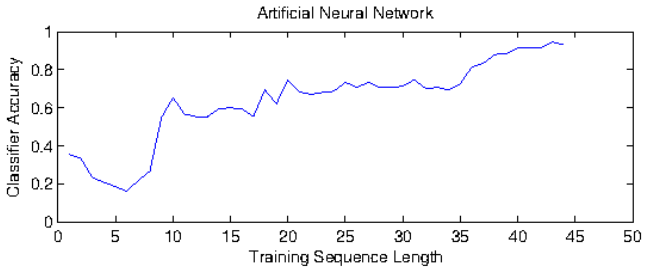


Figure 5: Neural network performance. Left: Error rates for different network configurations. Right: NN (3 input and 6 hidden nodes) learning rate for different length training sequences

The graph in Figure 5 (right) shows the learning rate of the neural network for different length of training sequences. The two networks, selected from the above experiment, have approximately the same performance. It can be seen in Figure 5 that the training sequence should be long enough to ensure an accurate prediction of the next symbol in the sequence. Using the test data, we are able to reach an accuracy of 91%, given a training sequence of 40 symbols. This is quite long, since in tennis there are approximately 70 serves per player.

It is striking how different these results are compared to the case study on serve-prediction discussed above. In order to explain these results, the next section performs extra data analysis on the two types of observed random behaviour.

6 Data Visualisation

Visual data analysis is important to observe frequencies in the recorded data in order to make a judgement about the randomness of a sequence. There are a number of techniques available to analyse time

series data. The distribution diagram for the placements of Thiemo and Michaella's serves are shown in Figure 6 and Figure 7 respectively. The diagrams were generated by successively scanning the sequence of serves with a window size of 10 symbols. Every pixel column of the stacked graph is the count of the last 10 choices. It can be seen that, in the beginning of the match, Thiemo plays very often to the

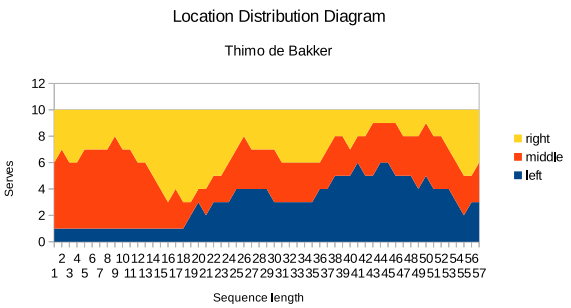


Figure 6: Distribution over all serves for Thiemo de Bakker

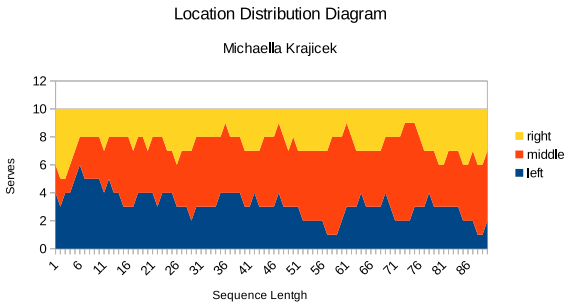


Figure 7: Distribution over all serves for Michaella Krajicek

middle of the opponents court. After about 14 serves, Thiemo starts to play to the right very often. We observe a change in strategy after about 20 serves since the amount of shots to the left and the middle of the opponents court are very close. For a short period of time he manages to distribute his serves almost equally among the three possible locations. The distribution diagram indicates that Michaella mixes up all 3 possible moves very well. Over the whole tennis match the number of serves to the possible locations are very close. However, this does not say much about predictability.

A Viz-Tree is a depth limited suffix tree that can be used to visualize frequencies in the data set. The sequence is pushed into the Viz-Tree where the frequencies are encoded as the thickness of the branches [7]. In case of a sequence that is truly random, all the branches of the tree have the same thickness. However, pseudo-random sequences generated by humans often show a tree with different sized branches. The Viz-Tree for one of our test sequences is shown in Figure 8 where it is obvious that some sequence are more likely to be observed than others, clearly identifying the data as not random.

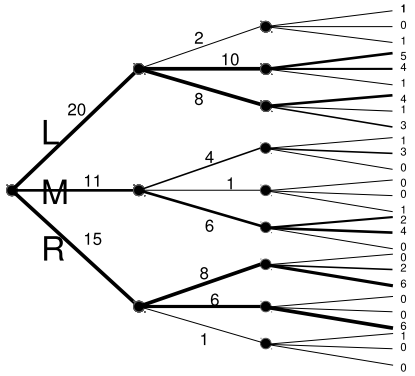


Figure 8: Viz-Tree for a pseudo random sequence generated by a non professional tennis player

When generating the same graphical representations of the serves for Thiemo and Michaella in Figure 9 and 10, the Viz-Trees look more like results expected from a truly random sequence. Both manage to distribute the serves reasonably well over all sequences of length 3, which gives a strong indication that these sequences are inherently harder to predict.

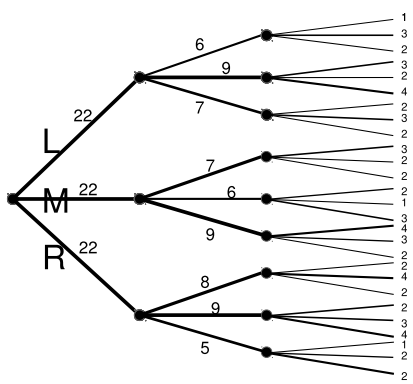


Figure 9: Viz-Tree for Thiemo De Bakker

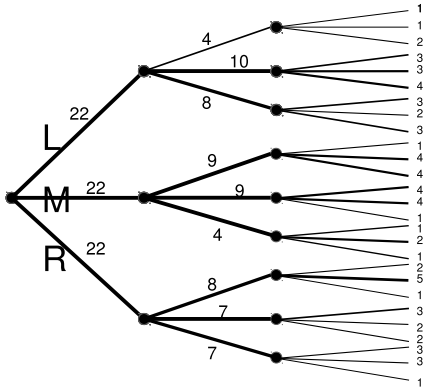


Figure 10: Viz-Tree for Michaella Krajicek

7 Conclusion

This paper presents an attempt to predict serving locations of professional tennis players. We compared the performance of Markov chains, prediction suffix trees and neural networks on serving data of Thiemo de Bakker and Michaella Krajicek. While the PST and NN performed best, predicting between 40% and 46% of the serve locations correctly, their performance stayed well below the initial expectations. Comparing these results to those for pseudo random data generated from volunteers, it is suspected that professional tennis players spend considerable effort obscuring their strategy. The obtained prediction accuracy is far from sufficient to provide the athlete with exploitable information during a professional tennis match. However, it could be used as a rough indication of where the opponent will serve next.

References

[1] G. Bejerano and G. Yona. Variations on probabilistic suffix trees: statistical modeling and prediction of protein families. *Bioinformatics*, 17(1):23–43, 2001.

[2] USPTA D. Fisher. Serving up the unpredictable, 2004. `http://addvantageuspta.com/default.aspx?act=newsletter.aspx&category=addvantage&Startrow=1&MenuGroup=ADD-depts&NewsletterID=484` (24.05.2014).

[3] O. Dekel, S. Shalev-Shwartz, and Y. Singer. Individual sequence prediction using memory-efficient context trees. *Information Theory, IEEE on*, 55(11):5251–5262, 2009.

[4] G.E.Hinton D.E.Rumelhart and R.J.Williams. Learning representations by back-propagating errors. 323:533 – 536, 1986.

[5] M. Figurska, M. Stanczyk, and K. Kulesza. Humans cannot consciously generate random numbers sequences: Polemic study. *Medical Hypotheses*, 70:182–185, 2008.

[6] F. S. Hillier and G. J. Lieberman. *Introduction to Operations Research*. McGraw-Hill, 2010.

[7] J. Lin, E. Keogh, S. Lonardi, J. P. Lankford, and D. M. Nystrom. Viztree: A tool for visually mining and monitoring massive time series databases. *VLDB 2004*, pages 1269–1272, 2004.

[8] S. Russel and P. Norvig. *Artificial Intelligence - A modern approach*. Prentice Hall, third edition, 2010.

[9] F. M J Willems, Y.M. Shtarkov, and T.J. Tjalkens. The context-tree weighting method: basic properties. *Information Theory, IEEE on*, 41(3):653–664, 1995.

Gesture Detection and Recognition by Repetition for Expressive Control

Bas Kooiker and Makiko Sadakata

Radboud University Nijmegen

Abstract

This paper presents the explorative research into gesture recognition on unsegmented three-dimensional accelerometer data. The application context is interactive dance and music performance. Repetition of gesture is used to distinguish between gesture and non-gesture using an algorithm for pitch detection which is adapted for multi-dimensional time-series called YIN-MD. Three gesture recognition algorithms, namely, Dynamic Time Warping (DTW), Phase shifting Dynamic Time Warping (DTW-PS), and Gesture Variation Follower (GVF) were evaluated with regards to orientation invariance, inter-user invariance, phase invariance and speed invariance. The evaluation indicated that DTW-PS is suited best for the purpose of this study.

1 Introduction

Gesture recognition methods have been used in numerous musical applications, ranging from interactive dance and music performance [5], music pedagogy [2] and control for a Digital Musical Instrument [1]. These applications typically use wearable three-dimensional accelerometer sensors for capturing human movement, which do not rely on the position of the performer. Applications which involve gesture recognition on accelerometer data usually mark the beginning and end of a gesture by using a button or footpedal press. This helps the system to reliably segment the signal. However, this approach constraints performers movements and this can be problematic, e.g. for a performance with choreograph (e.g., dance and theatre play). The goal of this project is to develop and implement a system which automatically detects the beginning and end of a gesture. This would replace the traditional manual coding of segmentation and thus expand the possibilities of gesture recognition as a tool for expressive performances. Using repetitive hand gestures made with three-dimensional accelerometer sensors, we compared the detection performance of three gesture recognition algorithms (see section 1.2 for detail information).

1.1 Repetition detection: YIN-MD

In music and expressive applications, repetition plays a fundamental role: repetitive rhythms or melodies in music, or recurring movements in dance are used to form the theme of a piece. Every repetition is described by a tempo. The first step towards automatic detection of tempo from multi-dimensional time series is to detect an interval at which repetition may be present. The second step is to determine whether the gestures at these intervals are similar enough to actually be repetition. Some of the pitch detection algorithms may be useful here, such as comb filters [14], Fourier analysis and auto-correlation [6]. Among these, we think that auto-correlation based method called YIN is promising because it is computationally efficient and has few tunable parameters. This method includes the following steps:

1. Autocorrelation function $r_t(\tau)$ (response over a range of possible delay values to detect periodicities)
2. Difference function $d_t(\tau)$ (making the method less sensitive for amplitude change)
3. Cumulative mean normalized difference function $d'_t(\tau)$ (deals with "too high" errors)

4. Absolute threshold on correlation strength

To extend this method to multi-dimensional time series, steps one to three were performed on the individual axis of the data stream resulting in a function $d'_{t,X}(\tau)$ where $X \in \{x, y, z\}$. These values were then combined by calculating the Euclidean distance between the three values. The fourth step was applied to detect gesture repetition (whether smallest $d''(\tau)$ is below a threshold At) and gesture length (smallest τ that for which $d''(\tau)$ is below At).

The algorithm has two tunable parameters: the window size W of the autocorrelation step, and the absolute threshold At for accepting or rejecting repetition.

1.2 Gesture recognition methods

The classical approach to pattern recognition is training a classifier with a large dataset of examples and then apply this classifier to new data. Hidden Markov Models (HMM) are very well suited for classifying temporal patterns [11] and the Rubine classifier [12] is also well known for gesture recognition.

Dynamic time warping (DTW)

In order to reduce training time and allow users to easily create their own gesture sets, template based classification methods were developed [16, 17]. The most common template based techniques use Dynamic Time Warping (DTW) combined with a nearest neighbor classifier. DTW is an algorithm which aligns two time series which may have a different length and speed and measures the distance between these two time series. DTW has been proven useful for accelerometer based gesture recognition [10].

Phase shifting dynamic time warping (DTW-PS)

DTW-PS [15] calculates the matching distance between the template and another time series x on P different phase shifts of the template. A phase shift here is defined as the transfer from a certain portion of the data samples from the start to the end of the series. ps is a value between 0 and 1 and $shift(ps, g)$ is the phase shifted template g with phase ps. The phase shift with the lowest distance is the most likely phase shift for that gesture template. Finding this most likely phase shift is done for every gesture template g in gesture set G. The recognized gesture in time series x is found as:

$$\operatorname*{argmin}_{g} \operatorname*{argmin}_{ps} DTW\,Distance(shift(ps, g), x)$$

Gesture variation follower (GVF)

So-called offline recognition techniques require a complete gesture trial before being able to classify it, such as DTW and DTW-PS. Bevilacqua (2010) developed a new approach to gesture recognition which focuses not only on recognizing the gesture, but also on the time progression of the gesture [2] and doing early recognition: recognizing the gesture before it is finished. This time progression means which exact part of the gesture is being performed at an exact time. This method, called gesture tracking rather than gesture recognition, was implemented in the Gesture Follower software, which has been proven effective for recognizing three-dimensional gestural data [1].

Caramiaux took the technique of gesture tracking even further by incorporating the tracking of gestural features in a gesture recognition model (Gesture Variation Follower, GVF). The gestural features that were incorporated in the model were phase, speed, scale and rotation. Whereas the Gesture Follower model was already able to track the phase and speed of a gesture performance, the GVF model could track even more specific features.

The GVF algorithm is based on Particle Filtering (PF) [13, 7]. A state model is defined which contains the current state of the system. This state is described by the gesture phase, speed, scaling and rotation. State samples (i.e. particles) are generated from the current model state distribution. These state samples are then weighted according to the every new observed sample. Recognition using this model is done by summing the weights of the particles for each gesture gesture template in a gesture set. The gesture with the highest total gesture weight is selected as the recognized gesture.

2 Algorithm evaluation

For the development of a gesture recognition system, we posed the following research questions: (1) whether YIN-MD is reliable enough for online repetition detection and (2) which gesture recognition method works best. The next section describes

2.1 Gesture vocabulary design

In order to validate the performance of YIN-MD and the three classification methods, we constructed a gesture vocabulary. There were three restrictions: (1) the gestures should be simple to remember and easy to perform, (2) the gestures should be continuously repeatable (i.e. the start and endpoint of the gestures are the same), (3) the gestures within a set should be sufficiently different so that it is possible to perform recognition. In order to overcome restriction one, a gesture vocabulary was constructed, based on simple geometrical shapes that were previously used in gesture recognition literature [10, 3]. Some shapes were discarded or modified to conform to the second restriction. The shapes included lines, triangles, squares, corners, rounded corners and circles. Of every shape there were multiple variations such as rotations and different directions. The total set contained 20 gestures.

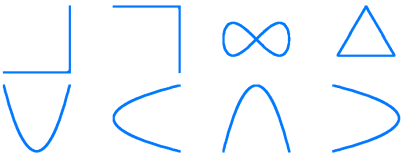


Figure 1: The selected gesture set of 8 gestures for the experimental research

We decided to use the classification performance of GVF as criteria because, among three algorithms, GVF was the only one which was known to be able to do online gesture recognition with early recognition. The most distinguishable set of eight gestures was defined using leave-one-out cross validation on the GVF recognition algorithm (see figure 1).

2.2 Data collection

Data was collected from ten participants (4 males, 5 females in the age range of 21 to 38 years old). Participants recorded the predefined gesture set 2.1 with three different hand orientations: 0, 45 and 90 degrees. Gesture data was recorded using a Sense/Stage MiniBee kit, equipped with three-dimensional ADXL345 accelerometers with a $\pm 16g$ range and 13 bit resolution at a sample rate of 33 Hz. A gesture recording application was implemented in Pure Data and Processing. A Pure Data application received annotated and recorded the accelerometer data, while a Processing application showed animations of the gesture on the screen for the subjects to follow.

The recording session consisted of three sets. For each gesture, the participants were first presented with an animation of a black dot moving along the trajectory of the gesture. Whenever they felt ready, they could start following the animation with their right hand and start recording the seven repetitive trials by pressing the space bar. The order of the eight gestures was randomized between participants. Between sessions, subjects were allowed to take a short break.

2.3 Gesture repetition detection

The YIN-MD algorithm runs on two parameters: absolute threshold (At) and tau (τ). The absolute thresholds determines the sensitivity of the algorithm to detected repetition. τ indicates the length of the auto-correlation interval.

Evaluation

The YIN-MD algorithm was evaluated on different parameter settings. To evaluate the performance of repetition detection of the algorithm for one parameter setting, simulations were run with the recorded gesture data. The data of each subjects $s_i \in S = \{s_1 ... s_{10}\}$ were evaluated sequentially such that there would be two consecutive repetitions of each gesture at a time. For every gesture $g_i \in G = \{g_1 ... g_8\}$ trial 1 and 2 were evaluated first. Then, for every gesture trial 2 and 3 were evaluated. This continued until the evaluation of trial 6 and 7 of each gesture g_i. With this method, every second repetition of

a gesture should be classified as repeating gesture, while every first repetition should be classified as non-repeating. The accuracy of the algorithm of detecting repetition correctly was used as performance measure.

Figure 2 shows the parameter optimization results of the YIN-MD algorithm. There is a clear interaction effect on the accuracy based on the two parameters. The maximum accuracy of .82 was acquired with $At = .10$ and $\tau = 10$. Higher values of τ decrease the average accuracy for different values of At. Higher values for τ also shift up the optimal value of At: for $\tau = 10$ the optimal value $At = .1$ while if $\tau = 60$, the optimal value $At = .4$.

A higher value of τ causes the algorithm to react too specific to cope with the variability of a performer, and therefor reject too many trials which are actually repetitions. This also explains why the optimal value for At goes up: a higher value for At results in a less sensitive behavior of the algorithm, therefor accepting more trials as repetition. This results in more non-repetition movement being classified as repetition, therefor increasing false positive errors.

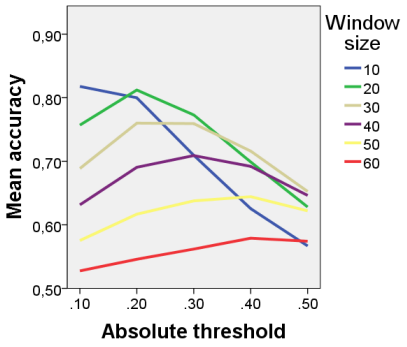


Figure 2: Parameter optimization of the YIN-MD algorithm

In performance arts, repetitions will often be made longer than two consecutive repetitions. We therefor evaluated parameter settings using seven consecutive repetitions instead of two. Optimal parameters were as follows: $\tau = 20$, $At = .3$ with an accuracy of .92. Longer repetitions increase the reliability. One of the explanations for this is that the transitions between repetition and non-repetition cause many of the errors.

2.4 Gesture classification comparison

GVF is a very flexible classification algorithm, able to do online frame-by-frame gesture classification as well as tracking of multiple expressive gestural features like phase, speed, scaling and rotation. The next section addresses the evaluation of this classification algorithm in different situation and how these capabilities compare to the capabilities of a benchmark gesture classification algorithm (DTW [10]) and an adaptation of this algorithm (DTW-PS). The first step was to reproduce the results that were presented in Caramiaux's study [3] to confirm that the setup of our project was comparable to the original project. After obtaining comparable results in the first experiment, the algorithms were evaluated and compared on (1) inter-user classification, (2) phase invariance and (3) speed invariance.

The GVF software is openly distributed online [4]. In this project, the same implementation and parameter settings were used as in the original paper. For the multi-dimensional DTW algorithm, an implementation was used presented in [9]. Small modifications were made to make the algorithm handle time series of different lengths and it was extended into the DTW-PS algorithm.

The performance evaluation followed the method used in previous studies [3, 10] which is the "leave-one-out" cross-validation. Every evaluation step, the classifiers are based on the i-th of 7 gesture trials and tested on the rest.

Sensor orientation invariance

For reproducing the orientation invariance results presented in [3], participants were asked to perform eight different gestures on three different orientations. The original experiment had a different gesture set and five orientations in the same range (0 to 90 degrees).

Following the arguments from Caramiaux [3], this sensor orientation invariance is supposed to solve the problem of users handling the sensor device differently (they used the accelerometer of a mobile phone to collect the user data). GVF would therefor give better results for users between sessions and also between users.

For each subject data, evaluation was performed with the three different orientations as a basis for the classifiers ($O_{tr} \in \{0, 45, 90\}$) and with the three orientations as the test data $O_{te} \in \{0, 45, 90\}$.

The recognition accuracy results were compared using three-way mixed design ANOVA analysis with Classifier (DTW, DTW-PS, GVF) as between subject factor and Orientation of the test data (0, 45, 90) and Orientation of the training data (0, 45, 90) as within-subject factors. These are shown in figure 3. Main effects of all factors were highly significant (Classifier: $F(2,24) = 57.95, p < .001, \eta_p^2 = .83$, Orientation test: $F(2,48) = 14.1, p < .001, \eta_p^2 = .37$, Orientation training: $F(2,24) = 33.63, p < .001, \eta_p^2 = .59$) with three-way interaction ($F(8,96) = 34.04, p < .001, \eta_p^2 = .739$).

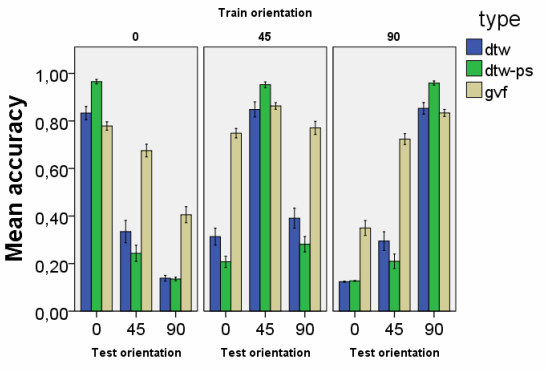


Figure 3: Mean classification accuracy for three algorithms in three orientations of test and training data

Simple effect analyses was conducted to contrast all classification rates, indicating that GVF performed significantly better ($p < .001$) than the other two algorithms in recognizing gestures at all cases where train and test orientation were not identical (e.g., train orientation is 45 degrees and test orientation is 90). In these cases, the classification performance between DTW and DTW-PS were not significantly different. When only looking at the cases where train orientation and test orientation were the same, DTW-PS performed significantly better than the two other algorithms ($p < .001$), with a less strong effect for train and test orientation 45: DTW-PS performed better than DTW ($p = .006$) and GVF ($p = .019$).

In summary, the above analysis revealed the following: GVF performed much better when recognizing rotated gestures as compared to the other two algorithms. The accuracy of GVF performance at 90 degrees rotation compared to the recorded template is very low and therefor not reliable to use for a real application, but the range up to 45 degrees rotation is much more reliable than the DTW-based algorithms.

Secondly, in the traditional gesture recognition setting (i.e. without difference in sensor orientation), the recognition accuracy of GVF was not better than the other two algorithms. Moreover, the DTW-PS algorithm clearly outperformed GVF with an accuracy of .95 against an accuracy of .82 respectively. Combined with the fact GVF is computationally much heavier than the other two algorithms, DTW-PS seems the be the best recognition algorithm choice when doing traditional gesture recognition. When incorporating rotation in a performance, GVF comes out on top.

Inter-user recognition

Now that we have seen that GVF indeed outperforms the other two algorithms when evaluating sensor orientation invariance, we will evaluate whether this indeed contributes to inter-user invariance i.e. training a model with data of one user and evaluating the data of another user. To evaluate this, cross-validation was performed between users combined with leave-one-out cross-validation on the trials. For each subject, the average accuracy was calculated over the validations when that subject was the train subject for a recognition model. Models from four of the nine subjects resulted in classification rates of about chance level (12.5 %). These subjects' data were excluded from this analysis as this seemed simply too different from the rest.

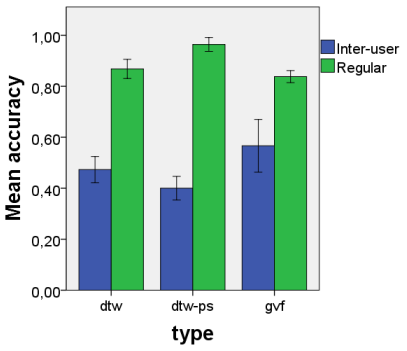


Figure 4: Mean classification accuracy for three algorithms on inter-user gesture recognition

The inter-user recognition results are presented in figure 4. The green bars show the accuracy when using the same train and test subject. The blue bars show the average accuracy when using a different train and test subject. The average recognition accuracy for the three recognition algorithms DTW, DTW-PS and GVF are .473, .400 and .566 respectively. ANOVA was used to analyze the difference between the classifiers. The

main effect was significant ($F(2, 15) = 8.905, p = .003, \eta_p^2 = .543$).

Pairwise comparison showed that GVF performed significantly better than DTW-PS ($p = .002$). The difference between GVF and DTW was marginally significant ($p = .097$). The difference between DTW and DTW-PS was not significant ($p = .251$). These results indicate that GVF might be the most reliable algorithm for inter-user recognition, but more research should be done to draw strong conclusions about this.

Phase invariance

For this specific project it is important that a gesture recognition algorithm is phase invariant i.e. the algorithm can recognize gestures if they are not started from the beginning. GVF does frame-by-frame gesture following and is capable of early recognition. DTW and DTW-PS is not capable of frame-by-frame gesture recognition. DTW-PS however, is especially adapted to combine the two characteristics of phase invariance and good recognition results in one algorithm.

Again, "leave-on-out" cross-validation was applied to evaluate the algorithms. This time, in every evaluation round, the test set was phase shifted a percentage $ps \in \{0, 0.05, 0.10...0.95\}$. Phase shifting is done by moving the annotations in the gesture data back, such that the start and end points of the gestures are shifted.

The results of the evaluation of phase invariance is shown in figure 5. The accuracy of GVF-PS is clearly higher than the other two. The difference between GVF and DTW is a bit more subtle. The average accuracy for DTW, DTW-PS and GVF are .335, .913 and .435 respectively. ANOVA analysis showed a strong significant difference between the three classifiers ($F(2, 24) = 809.5, p < .001, \eta_p^2 = .985$). Pairwise comparison showed that DTW-PS performed significantly better than the two other algorithms ($p < .001$) and GVF performed better than DTW ($p < .001$).

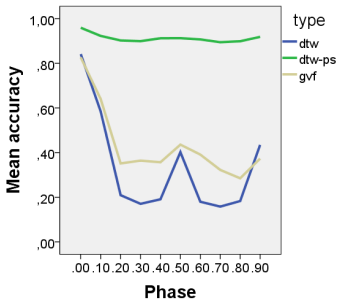


Figure 5: Mean classification accuracy for three algorithms on different phase shifts

DTW and GVF both showed a distinct peak in accuracy at $ps = 0.5$. This may be due to the fact that a number of the gestures are somewhat symmetrical in nature. Figure 7a shows the accelerometer data of the *curve down* gesture. The gesture has a symmetrical shape on two of the dimensions and a varying shape on the third dimension.

The DTW-PS has the best accuracy on $ps = 0$ with only a small decrease in accuracy on other phase shift values. This decrease in accuracy may be explained by artifacts due to the phase shifting of the templates. Figure 7b shows such an artifact. In the middle of the shifted data, there is a small drop of the yellow line and a small nudge up in the blue line. These are not part of the gesture but artifacts due to variability of gesture performance (i.e. the start and end of the gesture are not exactly the same). These artifacts are bigger when variability of a performer is bigger and this may influences the recognition accuracy.

Speed invariance

GVF tracks the speed of a performed gesture, therefor it should also be able to recognize gestures performed at different speeds. Being able to recognize gestures at different speeds is important for artistic applications because movement speed greatly influences a performers expression. Evaluation of this property was done by the same "leave-one-out" cross-validation method. Just as the evaluation of phase invariance, models were trained on the original gesture data, and the test set was manipulated to simulate differences in gesture speed. The test data was linearly interpolated and resampled with different intervals to simulate gesture speed from a factor $f \in \{0.8...1.33\}$ times the original gesture speed.

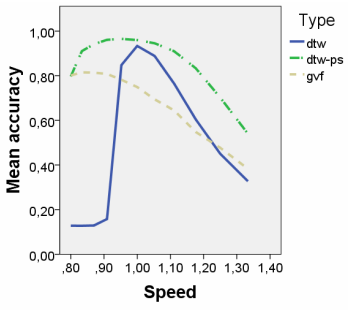


Figure 6: Mean classification accuracy for three algorithms on different speed factors

The results of the evaluation of speed invariance is shown in figure 6. The average accuracy over the different gesture speeds for DTW, DTW-PS and GVF are .488, .859 and .677 respectively. ANOVA showed a strong main effect between the classifiers ($F(2, 24) = 229.6, p < .001, \eta_p^2 = .950$) and pairwise comparisons showed highly significant differences between all the classifiers ($p < .001$). DTW-PS is the clear winner here. The figure shows that over all the different speeds, DTW-PS performs equally accurate or more accurate than both the other algorithms. DTW has a distinct peak in accuracy on speed factor $f = 1$, i.e. the original speed. In both directions, accuracy quickly drops. GVF has a less distinct accuracy peak and actually performs better in situations where $f < 1$ than where $f = 1$. When using GVF, it is better to record the gesture templates a bit quicker than the gesture speed how they will be performed.

3 Discussion

This study presented a method for gesture recognition of repeated gestures in multi-dimensional time series. The extended YIN-MD algorithm for repetition detection works well and fast enough for performance situations. The algorithm is not too sensitive to parameters settings in that it does not require calibration between users or sessions. After testing and evaluating the algorithm with the wearable sensors, the algorithm was also used in an application with Kinect for tempo detection of conducting movements with the same, reliable results. To further explore the possibilities of repetition detection in multi-dimensional time series, the applied approach of combining repetition strength results of individual time series into one measure of repetition strength could also be applied to other algorithms, such as fast Fourier transform or comb filter based techniques.

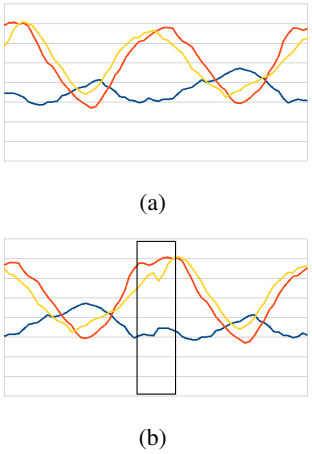


Figure 7: (a) Three-dimensional accelerometer data of the *down curve* gesture (bottom left in figure 1) and (b) the same gesture data shifted with phase shift $ps = .5$. In the black rectangle in figure b the artifact of phase shifting is visible. Time of two seconds on the x-axis and acceleration on the y-axis.

The results of the gesture recognition method comparison revealed two issues. When performing regular gesture recognition without intended gesture variation such as gesture rotation or sensor rotation, DTW-PS was the most reliable method. The phase shifting mechanism in the algorithm makes up for some of the timing mistakes of the users and therefor increases the accuracy compared to the original DTW algorithm. The DTW-PS algorithm was suggested again to be the best choice when phase and speed invariant performance was required. However, it is good to note that our results may be limited to evaluation of short and simple gestures. Further research is necessary for testing the generality of the current findings with e.g., more complex or longer gestures, gestures with more varying lengths, or gesture sets of different size.

When sensor orientation invariance or inter-user invariance is required, the GVF algorithm was the most reliable recognition method, though it did not perform significantly better tha DTW for inter-user recognition. The recognition accuracy of GVF with large sensor orientation differences as well as with inter-user recognition was not very high, definitely not high enough for most practical applications. Nevertheless, GVF showed the most potential for improvement in those contexts. GVF was implemented exactly in the same manner as in the original presented paper, which was optimized for sensor orientation invariance. Different parameter settings (i.e. the particle spreading parameters that influence speed and phase tracking) may improve accuracy of GVF in different contexts. The sensitivity to parameter setting of GVF makes it a very flexible algorithm, but it also makes it difficult to tune.

Our initial goal was to create a sophisticated system to use in interactive dance and music performance, where all the expressive control is done through physical movement of ones body rather than physical movement of an external controller. We believe that our comparative evaluation of three algorithms contributed to advance the gesture recognition in performance arts. An OSC based C++ application integrating the presented software in one package can be found at [8].

References

[1] Ruud Barth. Hand Gestural Control of Sound : a Digital Musical Interface. M.Sc Thesis, Radboud University, Nijmegen: Netherlands., 2011.

[2] Frédéric Bevilacqua, Bruno Zamborlin, Anthony Sypniewski, Norbert Schnell, and Nicolas Rasamimanana. Continuous Realtime Gesture Following and Recognition. *Gesture in embodied communication and human-computer interaction. Springer Berlin Heidelberg*, pages 73–84, 2010.

[3] Baptiste Caramiaux. *Studies on the Relationship between Gesture and Sound in Musical Performance.* PhD thesis, University of Paris VI, 2012.

[4] Baptiste Caramiaux. Gesture variation follower (gvf). Retrieved from: `http://www.baptistecaramiaux.com/gesture-variation-follower-gvf/`, 2014.

[5] Baptiste Caramiaux, Frédéric Bevilacqua, and Atau Tanaka. Beyond recognition: using gesture variation for continuous interaction. *Extended Abstracts on Human Factors in Computing Systems (CHI)*, pages 2109–2118, 2013.

[6] Alain de Cheveigne and Hideki Kawahara. YIN, a fundamental frequency estimator for speech and music. *The Journal of the Acoustical Society of America*, 111(4):1917, 2002.

[7] Doucet, Arnaud, Nando De Freitas and Neil Gordon. *An introduction to sequential Monte Carlo methods*. Springer New York, 2001.

[8] Bas Kooiker. GestureClassifier. Retrieved from: `https://github.com/baskooiker/GestureClassifier`, 2014.

[9] Daniel Lemire. Faster retrieval with a two-pass dynamic-time-warping lower bound. *Pattern recognition*, 42(9):2169–2180, 2009.

[10] Jiayang Liu, Zhen Wang, Lin Zhong, Jehan Wickramasuriya, and Venu Vasudevan. uWave: Accelerometer-based personalized gesture recognition and its applications. *International Conference on Pervasive Computing and Communications*, 42(9):2169–2180, 2009.

[11] Sushmita Mitra and Tinku Acharya. Gesture Recognition: A Survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 37(3):311–324, 2007.

[12] Dean Rubine. *Specifying gestures by example*, volume 25. ACM, July 1991.

[13] M. Sanjeev Arulampalam, Simon Maskell, Neil Gordon, and Tim Clapp. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Transactions on Signal Processing*, 50(2):174–188, 2002.

[14] Eric D. Scheirer. Tempo and beat analysis of acoustic musical signals. *The Journal of the Acoustical Society of America*, 103:588–601, 1998.

[15] Lambert Schomaker, Louis Vuurpijl, and Edward de Leau. New use for the pen: outline-based image queries. In *International Conference on Document Analysis and Recognition (ICDAR)*, pages 293–296. IEEE, 1999.

[16] Jacob O. Wobbrock, Meredith Ringel Morris, and Andrew D. Wilson. User-defined gestures for surface computing. *International Conference on Human Factors in computing systems (CHI)*, page 1083, 2009.

[17] D. Zhen, H. L. Zhao, F. Gu, and A. D. Ball. Phase-compensation-based dynamic time warping for fault diagnosis using the motor current signal. *Measurement Science and Technology*, 23(5):55601, 2012.

Event detection in Twitter: A machine-learning approach based on term pivoting

Florian Kunneman [a] Antal van den Bosch [a]

[a] *Centre for Language Studies, Radboud University*
P.O. Box 9103, NL-6500 HD Nijmegen, The Netherlands

Abstract

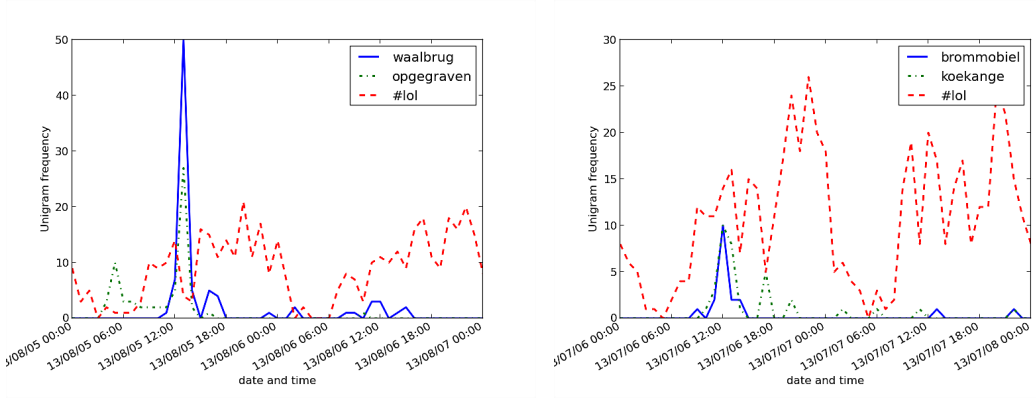
The large number of messages on Twitter posted each day provide rich insights into real-world events and public opinion. However, it is difficult to automatically distinguish tweets referring to such events from everyday chatter, and subsequently to distinguish significant events affecting many people from insignificant events. We apply a term-pivot approach to event detection from the Twitter stream. In order to filter out noisy and mundane events, we train a machine learning classifier on several rich features, and rank the events based on classifier confidence. After training and re-training the classifier using manually annotated data, we obtain an $F_{\beta=1}$ score of 0.79. However, a baseline that only takes into account the frequency of the tweets that refer to an event yields a better $F_{\beta=1}$ score of 0.86. We argue that performance is highly related to the definition of what makes a significant event, and that human understanding of this concept is not uniform.

1 Introduction

Microblogging platforms such as Twitter give users a voice to share ideas, opinions, and experiences with friends and the general public. Owing to the large user base on Twitter, the platform provides real-time information about what happens in the world. Detecting events and harvesting references to them from Twitter is therefore a highly valuable goal. However, this task is hampered by the nature and dynamics of Twitter. While news media select newsworthy items to write about, there is no such top-down selection process in the Twitter ecosystem. Events of public interest and mundane, insignificant events may both be characterized by bursty peaks in the usage of a set of terms in Twitter.

To give an impression of term burstiness in Twitter, consider the two examples in Figure 1. Example (a) displays the event of an excavation near the bridge 'Waalbrug' in Nijmegen, represented by a single joint rise and fall in the usage of the words 'waalbrug' and 'opgegraven' (Dutch for 'excavated') in Twitter. As a comparison, we also plot the frequency of the frequently used hashtag '#lol' in the same time window, which does not show any burstiness. It could be hypothesized that the first two terms both refer to an event, and possibly to the same event. Example (b) shows a similar pattern for the terms 'brommobiel' and 'koekange', peaking at about the same point in time, contrasted again with the non-bursty hashtag '#lol'. Without any additional knowledge, a system that leverages term burstiness might label the joint peaks in both examples as an event. However, further inspection shows that the peaks in example (b) denote a news report about a criminal act in the place of Koekange and an unrelated traffic accident with a scooter. A proper event detection system needs to filter out such insignificant events, possibly by taking into account additional features beyond burstiness.

The aim of this research is to expand existing work on detecting significant events on Twitter. We build on an approach proposed by [10]. They implement the *Twevent* approach to event detection in Twitter [7], and expand it by training a classifier on several features of an event to recognize significant events in contrast to mundane, insignificant events. We reproduce their experimentation and apply it to two months of Dutch tweets.



(a) (b)

Figure 1: Illustration of bursty and non-bursty term occurrences. Left: 'waalbrug' and 'opgegraven' (bursty) and '#lol' (non-bursty); right: 'brommobiel' and 'koekange' (bursty) and '#lol' (non-bursty).

2 Related Work

The detection of events in Twitter has been the goal of many studies. It is mainly approached as a clustering problem, with burstiness as the most important characteristic to detect an event. The most salient dichotomy among approaches is what [5] call *document-pivot clustering* and *term-pivot clustering*: burstiness is either measured at the level of tweets that share common terms, or at the level of single terms that display a joint burstiness over time. We provide an overview of the most important event detection systems, and summarize the performance on retrieving significant events reported by these studies.

2.1 Document-pivot clustering

The clustering of documents for the detection of events originates from the Topic Detection and Tracking (TDT) area of research [1]. Given a stream of news messages, any incoming message is linked to an existing event cluster or is the start of a new event cluster. [9] propose an adaptation of this approach to fast text streams such as Twitter. Incoming messages are either linked to an existing cluster, or grouped into a new one dependent on the distance to their nearest neighbour. Events are distinguished from other clusters based on the growth rate of a cluster. [9] obtain an average precision of 0.34 of retrieved event tweets versus tweets not related to an event, or spam. [8] reproduce the approach of [9], resulting in the retrieval of 1,340 events in 28 days of tweets, of which 382 (28%) are found to be significant.

Instead of clustering incoming tweets based on their raw content, alternative approaches focus on specific aspects of tweets that refer to future events. [11] state that important events on Twitter, in comparison to mundane events, have a common point in time to which multiple tweets refer explicitly. They extract events by clustering tweets that refer to the same point in time and mention the same entity. When ranking events based on the strength of the association between their date and entity, [11] obtained a P@100 (precision within the top-100 events) of 0.9 and a P@1000 of 0.52.

Yet another way to cluster tweets into events is to apply Latent Dirichlet Allocation (LDA) [2], by which individual words are linked to a topic based on their co-occurrence with other words. To detect bursty topics in Twitter, [4] build on the Twitter-tuned LDA implementation by [15], and expand it by adding topic distributions per time window and per user. Bursty topics are typically detected as a set of tweets from different users that contain similar words within a time window. A disadvantage of LDA is its dependence on parameter settings such as the number of topics, and the large number of sampling iterations that are required, leading to an extensively long duration for large sets of tweets. [4] set the number of topics to 30 in a period of 91 days, and obtained a precision of 0.76 for these topics (a precision@5 of 1.0).

2.2 Term-pivot clustering

[5] propose term-pivot (or feature-pivot) clustering as an alternative to document-pivot clustering for event detection from a news stream. Its two main advantages are the independence from parameter settings, and the event summary that is readily given by clustered terms. The first effective application of term-pivot clustering to event detection on Twitter is proposed by [14], who capture the burstiness of words by approaching them as signals and applying wavelet analysis to them. They obtain a precision of 0.76 for 21 events retrieved in a month of tweets from a Singapore user base.

[7] argue that multi-word segments or word n-grams, rather than single words, are beneficial both for the interpretation of an event and the detection of significant events. At the core of their *Twevent* system is the extraction of meaningful n-grams from tweets. N-grams are scored by their burstiness, and bursty n-grams are clustered into candidate events. The significance of a candidate event is dependent on the *Newsworthiness* of the individual n-grams, formulated as the combined chance of any n-gram sub-phrase to occur as an anchor text in Wikipedia, and the mutual similarity scores between the n-grams. [7] obtained a precision of 0.86 for 101 detected events on the same dataset as [14].

For the works discussed above, event significance is scored by an intuitive measure, such as the number of cluster terms [14] or the growth rate of a cluster [9]. Aiming to improve over these simple estimations of event significance, [10] apply *Twevent* to 15 days of English tweets and annotated the 4,249 resulting clusters as 'True news event' or 'False news event'. The clusters are linked to 15 rich features presumed to be indicative of their significance (these features are described in more details in Section 3.3.1). A classifier is trained and tested through 10-fold cross validation on all event clusters, resulting in a precision of 0.84 on 146 retrieved events, compared to 0.76 on 107 events by the original *Twevent* system.

In the study described here we adopt the approach by [10]. Where [10] build on the framework of *Twevent* to form clusters of segments, we base this clustering on unigrams rather than on segments. The rationale behind this is that in Dutch, the language we work with, word formation is characterized by compounding, which means that Dutch unigrams to a certain extent capture the same information as English bigrams. Compare, for instance, 'home owner' to 'huizenbezitter'.

As a definition of what makes an event significant, we follow the definition given by [8]: 'Something is significant if it may be discussed in the media.' As a proxy, we borrow the idea of [7] to include the presence of a certain name or concept as an article on Wikipedia as a weight in determining the significance of the candidate cluster of terms.

3 Experimental Set-up

3.1 Data

We collected two months of Dutch tweets by means of `twiqs.nl`, an archive of Dutch tweet IDs from December 2010 onwards [12]. The tweets in `twiqs.nl` are collected continuously from the Twitter API on the basis of a seed list of Dutch words and a list of the most active Dutch users. The harvesting is limited.[1] We collected the available tweets from 2013/06/22 until 2013/08/22, and filtered out non-Dutch tweets according to the language identification offered by `twiqs.nl`, resulting in a set of 65.02 million tweets.

3.2 Event detection

Our event detection approach takes the following steps.

3.2.1 Unigram selection by burstiness

To select candidate unigrams we first tokenize the tweets with ucto,[2] remove punctuation and user names, and lowercase the remaining words. Additionally, we remove stop words from each tweet. For each unigram we generated a time sequence of the tweets that contain the unigram. Following [7] we set the window size for this sequence to 24 hours, focusing on events that occur within a day.

[1] `twiqs.nl` harvests an estimated half of all Dutch tweets.

[2] `http://ilk.uvt.nl/ucto`

Given a day-by-day sequence of counts for a unigram, we score its burstiness per day by applying the state automaton approach to burstiness detection [6]. Each day a unigram can take on a bursty or normal state. The most likely sequence of states for a unigram can be uncovered by applying a Hidden Markov Model on the observed probability at each stage and the transition probability from state to state. We base the modeling of these two probabilities on the implementation by [4]. The observed probability of a count is based on a Poisson distribution for each state, which is defined as follows:

$$p(f_{ut} \mid v_t = l) = \frac{e^{-\mu_l} \mu_l^{f_{ut}}}{f_{ut}!} \tag{1}$$

Where f_{ut} is the frequency f of unigram u for time window t, l is either 0 or 1, and the normal and bursty states are denoted by μ_0 and μ_1, respectively. Following [4], we set μ_0 to the average count of a unigram over time and we set $\mu_1 = 3\mu_0$, i.e. an observed frequency has a higher probability to represent a bursty state when it approximates three times the average count. Also following [4] we set the transition probability σ_0 to 0.9 and σ_1 to 0.6, implying that a transition from a normal state to a bursty state is not very likely with a chance of 0.1. The chance that a bursty state reverts to a normal state is higher, with 0.4.

We use the Viterbi algorithm to dynamically find the bursty states for each unigram, and discard the unigrams without a bursty state as candidates. In our data set of 61 days, the method identifies $253,472$ bursty unigrams, with an average of $4,088$ per day ($\sigma = 703$).

3.2.2 Unigram similarity

To cluster unigrams into event clusters, we adopt the approach by [7]. For each day in our dataset, the similarity between all pairs of bursty unigrams is calculated and clusters are formed based on this similarity graph. To calculate the similarity, each time window t is divided into M sub-time-windows. Following [7] we set the size of M to 12 (i.e. two hours per sub-time-window). The similarity between any pair of unigrams u_a and u_b on a day is calculated as follows:

$$sim(u_a, u_b) = \sum_{m=1}^{M} w_t(u_a, m) w_t(u_b, m) sim(T_t(u_a, m), T_t(u_b, m)) \tag{2}$$

The sub-time-window similarity between unigrams is computed by collecting the tweets in which the unigrams are mentioned, and generating two pseudo-documents containing all concatenated tweets in which one or the other unigram occurs. Terms in these documents are weighted by $tf - idf$, and the cosine distance between the two pseudo documents is calculated as the similarity score between the two unigrams. This calculation favors pairs of unigrams that are mentioned with comparable content and that are most bursty in the same sub-window. Furthermore, it considers the similarity between tweets rather than the co-occurence of unigrams, which is reasonable given the shortness of tweets.

3.2.3 Term clustering

Given the similarity graphs of bursty unigrams per day that result from the previous step, unigrams are clustered into event clusters. Following [7], we apply Jarvis-Patrick clustering. This algorithm has two parameters, k and l. For any two unigrams to be clustered together, they have to occur in each others k-nearest neighbours and they have to share at least l common neighbours in their k-nearest neighbors. Advantages of this algorithm are its limited computational cost and the fact that the number of topics does not have to be defined.

[7] found that the l parameter is too restrictive for this task. Following them, we only took into account the k parameter and set $k = 3$, linking unigrams if they occur in each other's top-3 most similar unigrams. Unigrams that were not linked to any other unigram were discarded. As a result, we retrieved a total of $33,452$ event clusters from the 61 days of bursty unigrams (548 on average per day).

3.3 Event significance classification

Event significance classification can be seen as what [10] call 'event filtering'. The events that result from clustering are sorted into significant and insignificant events. We apply the same approach to

event filtering as [10]: describing event clusters by rich features and training a classifier to distinguish significant from insignificant events.

3.3.1 Features

In their research, [10] include 15 rich features. Most of the features that we use are adopted from [10]. We describe the features below, and make a distinction between cluster features and tweet features: respectively the characteristics of the unigrams that describe a cluster and the characteristics of the tweets in which the unigrams of a cluster occur on the day of their burstiness (referred to as event tweets). For each of the 15 features, we give a full name and an abbreviation between brackets, which will henceforth be used to refer to the feature.

Cluster features

- Unigrams (UNI) - the number of unigrams in the event cluster. Arguably, a cluster which is described by many unigrams is not likely to represent a coherent, significant event.
- Edges (EDGE) - the average number of clustering edges between the unigrams in the event cluster. This feature describes the density of a cluster.
- Similarity (SIM) - the average similarity score, as described in section 3.2.2, between unigrams in the event cluster.
- Burstiness (BST) - the average burstiness of unigrams in the event cluster, adopted from the bursty probability calculation in [7]. This probablity is based on the expected frequency $E[u|t]$ of a unigram u in a time window t, given its Gaussian distribution:

$$E[s|t] = N_t P_s = N_t * \frac{1}{l} \sum_{t=1}^{L} \frac{f_{u,t}}{N_t} \tag{3}$$

  Here, N_t is the number of tweets during day t, L is the number of time windows containing u, and $f_{u,t}$ is the frequency of u in time window t. Given $E[s|t]$, the bursty probability $P_b(s,t)$ is calculated as follows:

$$P_b(s,t) = S(10 * \frac{f_{s,t} - (E[s|t] + \sigma[s|t])}{\sigma[s|t]}) \tag{4}$$

  S is the sigmoid function and $\sigma[s|t] = \sqrt{N_t P_s (1 - P_s}$, the standard deviation of the Gaussian distribution.
- Newsworthiness (WIKI) - the average newsworthiness of unigrams, which is operationalized in [7] as the ratio by which terms that are (in) the title of a Wikipedia page are referred to from other pages from anchored links. Terms that have a high probability to be used as anchor to their page are believed to be more newsworthy. To calculate the newsworthiness score for all bursty terms, we downloaded a dump of the Dutch Wikipedia pages from November 14th 2013 (the closest date after our data set).[3]

Tweet features

- Document Frequency (DF) - the relative frequency of the event tweets, calculated as the number of event tweets on the given day divided by the total number of tweets on that day.
- User Document Frequency (UDF) - the relative number of different users that refer to the event, calculated as the number of users that posted one of the event tweets, divided by the total number of event tweets.
- Hashtags (HT) - the average number of hashtags (#) per event tweet
- URLs (URL) - the percentage of event tweets that contain a URL (any token starting with ('http://')
- Replies (REP) - the percentage of event tweets that start with a username (tokens that start with a '@'), which is typical of a reply.

[3] `http://dumps.wikimedia.org/nlwiki/20131114/`

- Mentions (MEN) - the percentage of event tweets that contain a mention of a username, on any position other than the start of a tweet.
- Cohesiveness (CHS) - the average number of unigrams in tweets. If the event tweets contain two or more of the clustered unigrams, they are more likely to refer to a cohesive event.
- Informativeness (INF) - the relative number of different words in the event tweets. Spam messages are often characterized by a narrow vocabulary, while events that arouse the attention of a lot of people might be referred to with a bigger variation of words.

3.3.2 Classification

While [10] annotate all 4,249 event clusters retrieved by the *Twevent* approach from their data set, we did not annotate all 33,452 event clusters retrieved from our data set. Instead we selected a subset of the data. To make sure we had enough significant events in this subset, we trained a classifier on 350 labeled event clusters on the first two days in our data set and applied it to the remaining days. The 1000 events of which the classifier was most confident were used as data set for our experimentation.

As classifier we made use of the SNoW implementation of Winnow [3][4]. This algorithm is known to offer state-of-the-art results in text classification, and outputs a per-class confidence score by which instances could be ranked. To tune the different parameters of Winnow (α, β, $\theta+$, $\theta-$, the number of iterations and the thick seperator), we applied a heuristic hyperparameter optimization scheme that makes use of wrapped progressive sampling on training data [13].

To obtain labeled data for the preliminary classification we ranked the event clusters in the first two days based on the average similarity score of their unigrams. One of the authors annotated the top 350 of these events as significant or not, resulting in 153 events labeled as significant and 197 events labeled as insignificant. The classifier was trained on these 350 labeled events and was applied to all events in the remaining days in the data set. The 1,000 events that were most confidently scored as significant by the classifier were used in our main experimentation.

To obtain trustworthy labels for the 1,000 events we asked 8 annotators to each label 250 events as significant or not. The data was split in a way that each event was annotated by two annotators, with 8 unique annotator pairs (125 events per pair). We presented them with a list of events represented by a date, the event unigrams, and a sample of 10 of the event tweets. In our explanation of the task, we gave them the definition of a significant event that we specified in section 2.2, as well as a few examples of typically significant and insignificant event clusters. The task was to annotate each event as either significant, insignificant, or doubtful. We additionally asked the annotators to indicate if the event was a social event, which we planned to use for additional research.

354 of the 1,000 event clusters were indicated by both annotators as significant, 723 were annotated as significant by at least one of the two annotators and 277 events were annotated by both annotators as insignificant. The mean inter-annotator agreement was fair ($\kappa = 0.25$, with a standard deviation of 0.11).

3.4 Evaluation

Given the 1,000 annotated event clusters, we evaluated classification performance by 10-fold cross-validation. We apply classification with a strict and lax labeling. For strict labeling, only events that were indicated as significant by two annotators are labeled as significant, while for the lax labeling, events that were annotated by one as significant are seen as significant. To score the performance, we calculate the precision, recall and F1 scores for the retrieval of significant events. As baselines we ran the classifier separately on the intuitively most effective features for significant event classification: burstiness (BST), the number of tweets mentioning the event (DF), and the similarity between unigrams (SIM).

4 Results

The results are given in Table 1. Both in the strict and the lax setting the classifier that bases its judgements on all feature values yields a worse performance than one of the classifiers based on a single

[4] http://cogcomp.cs.illinois.edu/page/software_view/SNoW

feature. In the strict setting, the relative document frequency is sufficient, while for the lax setting the term burstiness leads to a peak performance of .94.

	Strict			Lax		
	Precision	Recall	F1	Precision	Recall	F1
DF	.80	.95	**.86**	.73	.99	.84
SIM	.54	.93	.68	.84	.95	.89
BST	.57	.84	.68	.93	.94	**.94**
All	.76	.90	.79	.91	.93	.92

Table 1: Results for the classification of events as significant in the strict and lax setting, by performing classification based on a single feature (DF, SIM or BST) and based on all 13 features.

In Table 2 we show the five events that were most confidently ranked as significant by the classifier that uses all features. Three of the events are arguable significant for a large number of people: the football match 'Spanje-Italie', a goal of Clarence Seedorf, and a reference to the television program 'Miracle Run'. The next event is of a personal nature (school performance), while the final case is only arguably newsworthy (breeding insects for improving the environment).

date	event terms	example tweet
27-06-2013	spanje-italie, italie	spanje-italie kijken ik ben voor itali
15-07-2013	botafogo clarence	RT @433NL VIDEO Oud maar nog steeds gedreven Clarence Seedorf 37 scoorde vanavond een heerlijke goal voor Botafogo http://t.co/Wj7hp
15-07-2013	efron zac miracle	@BBergstra op rtl 8 Miracle run Gaat over een autistische tweeling met Zac Efron x
03-07-2013	bevorderd gymnasium	Bevorderd naar gymnasium 3 :-)
04-07-2013	milieuproblemen kweken	Insecten kweken als op lossing voor voedsel en milieuproblemen http://t.co/cRTFvL7ToT

Table 2: events classified as significant most confidently based on all 13 features in a 10-fold cross-validation setting.

5 Conclusion and discussion

We reproduced the term-pivot approach to event detection proposed by [7] and applied it to two months of Dutch tweets. In line with [10] we annotated the resulting events on their significance and trained a machine learning classifier based on 13 features. We found that the relative frequency by which an event is mentioned provides a sufficient cue to recognize significant events as opposed to feeding the classifier all 15 features, yielding $F_{\beta=1}$ scores of 0.86 and 0.79, respectively.

Our system obtains precision values that are similar to the ones reported by [10] (around 0.80), while our recall values are much higher. An explanation is that [10] train and test on a much larger set of 4,249 events with a smaller fraction of significant events, making the task more challenging. Furthermore, we train and test on the already ranked output of our classifier. As [10] we find that the number of event tweets and the number of URLs in these tweets are important features to recognize significant events. On the other hand, user document frequency (UDF), newsworthiness (WIKI) and similarity values (SIM), reported by them as useful features, did not have a big influence on the classifier performance in our experiment.

In our experiment, two annotators labeled each event. The low agreement value ($\kappa = 0.25$) shows that it is difficult even for humans to decide whether an event is significant or not. We found that it is not trivial to provide the annotators with an unambiguous definition of what makes a significant event. In future work we will attempt to develop a sharper definition.

Acknowledgements

This research was funded by the Dutch national program COMMIT. We thank Erik Tjong Kim Sang for the development and support of the `http://twiqs.nl` service.

References

[1] J. Allan, R. Papka, and V. Lavrenko. On-line new event detection and tracking. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '98, pages 37–45, New York, NY, USA, 1998. ACM.

[2] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[3] A. Carlson, C. Cumby, J. Rosen, and D. Roth. The snow learning architecture. Technical report, University of Illinois at Urbana-Champaign, 1999.

[4] Q. Diao, J. Jiang, F. Zhu, and E.-P. Lim. Finding bursty topics from microblogs. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 536–544. Association for Computational Linguistics, 2012.

[5] G. P. C. Fung, J. X. Yu, P. S. Yu, and H. Lu. Parameter free bursty events detection in text streams. In *Proceedings of the 31st international conference on Very large data bases*, pages 181–192. VLDB Endowment, 2005.

[6] J. Kleinberg. Bursty and hierarchical structure in streams. *Data Mining and Knowledge Discovery*, 7(4):373–397, 2003.

[7] C. Li, A. Sun, and A. Datta. Twevent: segment-based event detection from tweets. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 155–164. ACM, 2012.

[8] A. J. McMinn, Y. Moshfeghi, and J. M. Jose. Building a large-scale corpus for evaluating event detection on twitter. In *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management*, pages 409–418. ACM, 2013.

[9] S. Petrović, M. Osborne, and V. Lavrenko. Streaming first story detection with application to twitter. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 181–189. Association for Computational Linguistics, 2010.

[10] Y. Qin, Y. Zhang, M. Zhang, and D. Zheng. Feature-rich segment-based news event detection on twitter. In *International Joint Conference on Natural Language Processing*, pages 302–310, 2013.

[11] A. Ritter, Mausam, O. Etzioni, and S. Clark. Open domain event extraction from twitter. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '12, pages 1104–1112, New York, NY, USA, 2012. ACM.

[12] E. Tjong Kim Sang and A. van den Bosch. Dealing with big data: The case of twitter. *Computational Linguistics in the Netherlands Journal*, 3:121–134, 12/2013 2013.

[13] A. Van den Bosch. Wrapped progressive sampling search for optimizing learning algorithm parameters. In *Proceedings of the 16th Belgian-Dutch Conference on Artificial Intelligence*, pages 219–226, 2004.

[14] J. Weng and B.-S. Lee. Event detection in twitter. In *Proceedings of the AAAI conference on weblogs and social media (ICWSM-11)*, pages 401–408, 2011.

[15] W. X. Zhao, J. Jiang, J. Weng, J. He, E.-P. Lim, H. Yan, and X. Li. Comparing twitter and traditional media using topic models. In *Advances in Information Retrieval*, pages 338–349. Springer, 2011.

Case-Driven Agent-Based Simulation for the Development and Evaluation of Cognitive Architectures

Samer Schaat Dietmar Dietrich

Vienna University of Technology, Institute of Computer Technology, Gußhausstrasse 27-29/E384, 1040 Vienna

Abstract

This paper introduces a methodology for the interdisciplinary development and evaluation of cognitive architectures. We use a combination of casuistry, agent-based modelling as a scientific method, and use-case-driven requirements engineering to support interdisciplinary knowledge translation, while accounting for the complexity in describing and evaluating a cognitive architecture intended to represent the functioning of the human mind. A shared representation of exemplary cases demonstrating the requirements for developing a cognitive architecture serves as a platform and guideline for interdisciplinary collaboration. Model requirements are introduced by different disciplines following a case-based approach and concretized in deterministic simulation cases using use-case-driven requirement engineering. Agent-based simulation is harnessed to evaluate the model, enabling us to test our assumptions and the model's plausibility.

1 Introduction

Although Artificial Intelligence has led to innovative applications, developing information systems of human information processing and decision making are still open challenges. One reason is the underestimation of serious and regular *interdisciplinary collaboration* between the humanities and computer science using a *common framework*. But since the mind is the domain of multiple disciplines within the humanities and natural science, an intensive interdisciplinary collaboration seems indispensable for scientific progress. In such an approach, other disciplines provide insights into the mind. Computer technology provides a methodology for developing and testing a deterministic model of the human mind (i.e. the mind as a biological computer). Such an approach has recently been used in AGI (Artificial General Intelligence) to simulate the mind and focuses on the search for a holistic cognitive architecture able to generate human-like general intelligence. A synthetic approach is probably our best chance to understand the human mind and harness this knowledge in technical systems. Most of all, it allows us to test our ideas of how the mind works by running them as computer programs. The complexity of capturing the mind's functioning as well as interdisciplinary knowledge translation pose special challenges to be considered while developing a methodology for interdisciplinary development and evaluation. More precisely: Which method is suitable for translating findings from other disciplines into a deterministic cognitive architecture for a humanoid agent and evaluating the results? In this paper, we tackle this question and present *case-driven agent-based simulation* as a particularly suitable methodology to guide interdisciplinary collaboration towards finding the required functions and data for a simulation model of the human mind, which is then evaluated through agent-based simulation.

2 Related Work

As emphasized in [1], evaluation of cognitive architectures poses special challenges. The approach of the cognitive architecture ACT-R (Adaptive Control of Thought-Rational) is to validate their theories by associating the components of their model with specific brain regions and conducting fMRI studies [2]. Besides basic questions about how appropriate fMRI is for deducing knowledge about the brain's functioning, the lack of detailed understanding of the neural mechanisms of cognition makes validation of a cognitive architecture on a neural level unfeasible yet. Other approaches use variations of the Turing

test, e.g. a tournament [3] with a human jury, to decide how human-like the respective agents are. But such approaches only test the behavior; while the setting impedes the use of objective criteria and this general evaluation of human-likeness remains impossible. The abstract Newell test is proposed by [4] to evaluate cognitive architectures, in particular using conceptual criteria based on [5] (e.g. flexible behavior, real-time performance, adaptive behavior) for evaluation. While such criteria are beneficial for theory building and support model development, their suitability for evaluation purposes is questionable due to the lack of falsifiability [6]. In general, such tests do not present an objective possibility to apply these criteria. However, as put forth by [7], a cumulative research program is more appropriate for cognitive architectures based on the argument that cognitive architectures are not falsifiable as a whole, and that instead only individual assumptions about them are, since "in the absence of task knowledge, architectures make few if any predictions" [7, p.512]. Here, "theory development and the way in which a theory responds to predictive failures" [7, p.514] distinguishes science from pseudo-science. In this regard, Lakatos distinguishes core assumptions that cannot be falsified from falsifiable protective hypotheses which are the subject of advance and adjustment.

Agent-based simulations (ABS), which provide an approach for interdisciplinary model building and evaluation [8], can be regarded as following Lakatosian principles. However, [9] warns us that a simulation is no better than the assumptions built into it, and that a computer can do only what it is programmed to do. Nevertheless, simulations can provide new knowledge, but "…even when we have correct premises, it may be very difficult to discover what they imply…" [9, p.15]. This is particularly the case with the paradigm of agent-based simulation. In essence, ABS explains the global system's complexity at the macro level, e.g. an agent's behavior, with the interaction of many local components at the micro level, e.g. an agent's assumed and specified underlying processes which generate the behavior [10]. These generative micro-specifications are usually formulated by other disciplines, which is to be regarded as a novel kind of interdisciplinary collaboration [8]. In this regard, ABS is termed a third way of performing science [11]. When using ABS, validation may occur at both the micro and macro level [10], thereby implying two separate types of falsifiability: the first questions the validity of the underlying processes, while the second analyzes whether the generated behavior matches observed behavior, i.e. whether the model results fits the empirical data. This happens precisely when the micro-specifications generate the macro behavior in simulations. However, such validation of ABS using human-related empirical data is difficult to achieve, as shown in an examination on a sample basis in [12], where most ABS simulating human behavior could not be empirically validated. This is mainly due to difficulties in data acquisition on a psychological and sociological level. Despite the lack of empirical validation, simulation of an agent-based model shows its plausibility by validating its functioning and robustness. In addition to model development and evaluation, ABS may guide empirical research by providing a theory-building tool [8]. Hence ABS also allows us to ask and test new questions, and provides possible counterintuitive insights. The use of cognitive architectures as a decision unit for an agent instead of simplified decision rules is still in its infancy, as the approach forces a more detailed model which in turn requires more interdisciplinary collaboration and thus more challenging requirement analysis and knowledge translation.

3 From Exemplary Cases to Structured Simulation Cases

To analyze the requirements of a holistic cognitive architecture, we use the expertise of psychoanalysts and neuroscientists in a casuistic approach, where the case of how a hungry agent might behave in the presence of another agent and a food source is described in a narrative form called an *exemplary case*. To be able to *formulate explanatory assumptions* of the described behavior and extract requirements for a simulation model thereof, a *representative case* that demonstrates specific aspects (of phenomena) is described in an exemplary fashion. It must be emphasized that the expertise of specialists (such as psychoanalysts and neuroscientists) with regard to real-world conditions is necessary to define the requirements for a human-inspired cognitive architecture. Our experience in interdisciplinary collaboration shows that structuring and concretization are needed to extract these requirements from the often non-deterministic psychoanalytic and neuroscientific descriptions. These criteria are taken into account in the conceptualization of our methodology. The coarse procedure in such *case-driven agent-based simulatio*n (see Fig.1) consists of (1) a narrative description of an interdisciplinary exemplary case which incorporates theories describing the phenomena to be explained; (2) analyzing the exemplary case and transforming it into a *structured simulation case* from which concrete requirements are extracted; (3) developing algorithms and knowledge representations that fulfill the requirements; (4) implementing

functions which generate – and parameters which determine – the behavior described in the exemplary case. The resulting model is evaluated by observing whether a parameterized simulation of the model results in the expected behavior, and in particular whether the parameters specified in the simulation case generate the predicted behavior. If not, an analysis is required to identify which step of the interdisciplinary methodology must be adapted in the next iteration.

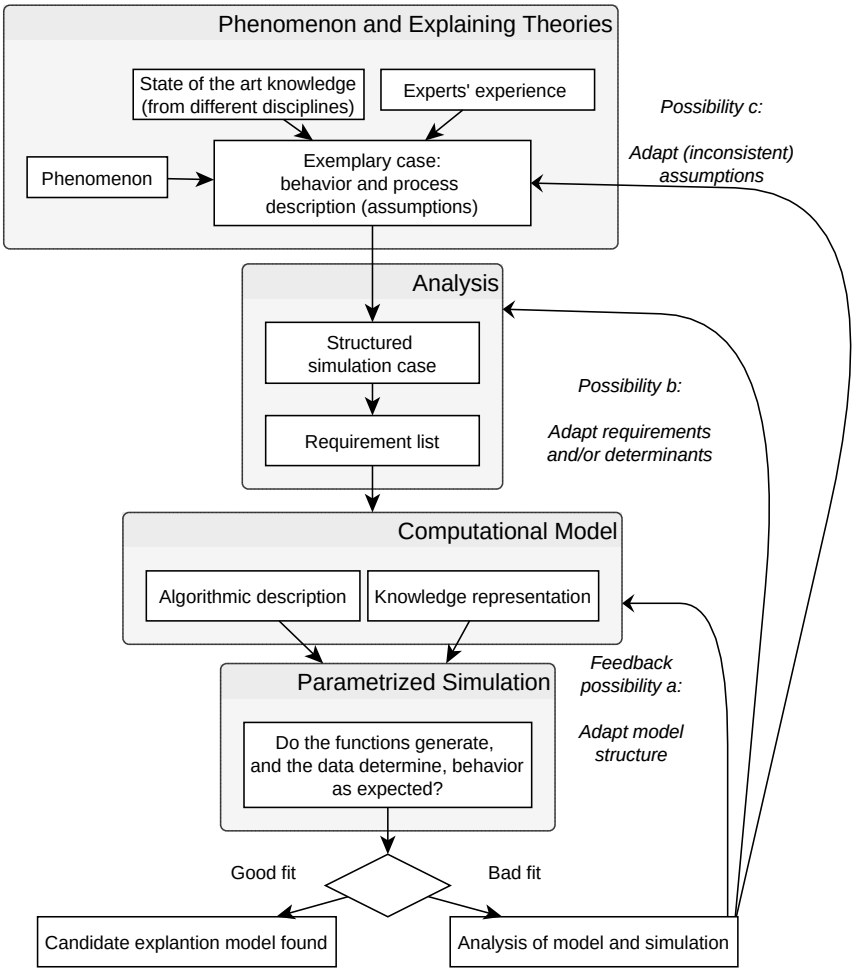


Fig. 1: Overview of case-driven agent-based simulation.

3.1 Exemplary Case

Many disciplines, such as medicine and law, use typical cases from their practice to demonstrate specific concepts. This vivid form of description often supports analysis and discussion. In developing and evaluating a cognitive architecture, we use case descriptions to exemplify research questions (e.g. regarding the foundations of a cognitive architecture) – including questions from psychology (e.g. motivation and decision making) and sociology – and demonstrate how our hypotheses answer them. Above all, the exemplary case serves as a foundation for creating a platform between different disciplines. Since it is a common form of description for psychoanalysts, psychologists, and neuroscientists, we use it as a point of departure for interdisciplinary exchange and knowledge translation. Given specific topics and research questions, researchers from different disciplines exemplify the topics at stake in a narrative form. This procedure not only supports knowledge translation, but also the analysis of research questions, demonstration of assumptions, and the explication of requirements to answer these question.

Hence an exemplary case is primarily a description of concrete phenomena to be explained by answering a research question. In particular, it describes the situation and events in a concrete case. To consider a phenomenon holistically, not just one specific decision in a case is described, but alternative courses of action that are relevant to understanding the phenomenon are as well.

To analyze the requirements of basic human decision making, we choose an exemplary case describing how a hungry agent behaves in a situation featuring another agent and a food source. Possible courses of action are eating or sharing the food source, fighting with the other agent, or leaving the scene.

As a first step, the behavior of agents under specific conditions (environment and inner state), i.e. a behavior model, is described. This includes a description of their personalities. Next, experts analyze this behavior and – based on their experience of real world conditions and state of the art knowledge – formulate a process model which consist of describing basic psychic processes. Generally, these descriptions state base assumptions for the model to be developed and must be evaluated in the course of the simulation. The assumptions are plausible, since they are based on evidence from state of the art literature and/or experts' experience, but will nevertheless be validated in simulations. However, in a Lakatosian sense (see Chapter 2) core assumptions that cannot be falsified by this methodology exist (e.g. about the suitability of involved disciplines).

An exemplary case helps to avoid drifting into abstract discussions, but also tends to be indeterministic and inconsistent, as well as often having gaps in its assumptions. To be able to concretize and specify the requirements and model, we need to transform the exemplary case into a structured and deterministic form of description, the so-called simulation case, which is an appropriate development and evaluation tool for simulation. The simulation case serves as a consistent and structured shared description of the case for all involved disciplines.

3.2 Clarifying the Exemplary Case

During the transformation of the narrative exemplary case into a structured simulation case, some difficulties stemming from different methodologies in the various disciplines became evident. This led to the definition of criteria that such an exemplary case must fulfil in order to be transformed into a structured description for a deterministic model. The primary criteria are the explication of the theories' assumptions and their consistent and deterministic description. It is this step which enables use of the complete exemplary case for a computer simulation. Two important stages are distinguished: First, clarification of the unstructured narrative exemplary case and second, transformation of the exemplary case into a deterministic and structured simulation case. The clarification step results in an initial structuring using the following template:

Title: Agent Adam is hungry.
Question: What are the basic mechanisms and determinants of human decision making and how must an according cognitive architecture look like?
Initial state: *Description of agents and environment.*
Case description: Narrative description of agent Adam's behavior when confronted with another agent, a food source and its own bodily needs, and assumptions about the corresponding psychic processes.
Possible actions: Eat, Share, Give, Fight, and Go away.

Such an exemplary case allows the transformation into a deterministic simulation case, which in turn permits the specification of functions and according parameters capable of generating the described behavior in simulations. Only such a case description allows the construction of a causal chain that leads to the specified behavior, a key requirement for developing a functional model.

3.3 Simulation Case Structure

After clarifying the narrative exemplary case we transform it to a *structured simulation case*, sketched in Fig.2[1]. One focus is to analyze the data that determine behavior; in particular, how a change in these *determinants* causes a change in the agent's behavior. Generally speaking, four groups of determinants are specified: personality parameters, memory, environment and the agent's internal state (represented by the psychoanalytic drive concept [13] and Damasio's emotion concept [14]). This analysis of data processing supports the development of functions that generate behavior.

The *simulation case framework* (see Fig. 2) represents the abstract guideline which is concretized in the respective scenarios. The standard scenario describes the behavior in detail, and the concrete determinants (e.g. concrete valuated memories and bodily needs) that lead to this behavior according to the assumptions of the exemplary case are specified. The standard scenario's provides a list of the determinants, a structured and causal description of how they determine behavior, the decided action and

[1] Details cannot be provided due to lack of space. Fig. 2 serves to give an impression about a simulation case.

the agent's final state (given by emotions and drives). The simulation case's general requirements are all included in the standard scenario; hence the description of alternative scenarios consists primarily of information about how specific determinants should be changed to achieve a change in the agent's decision.

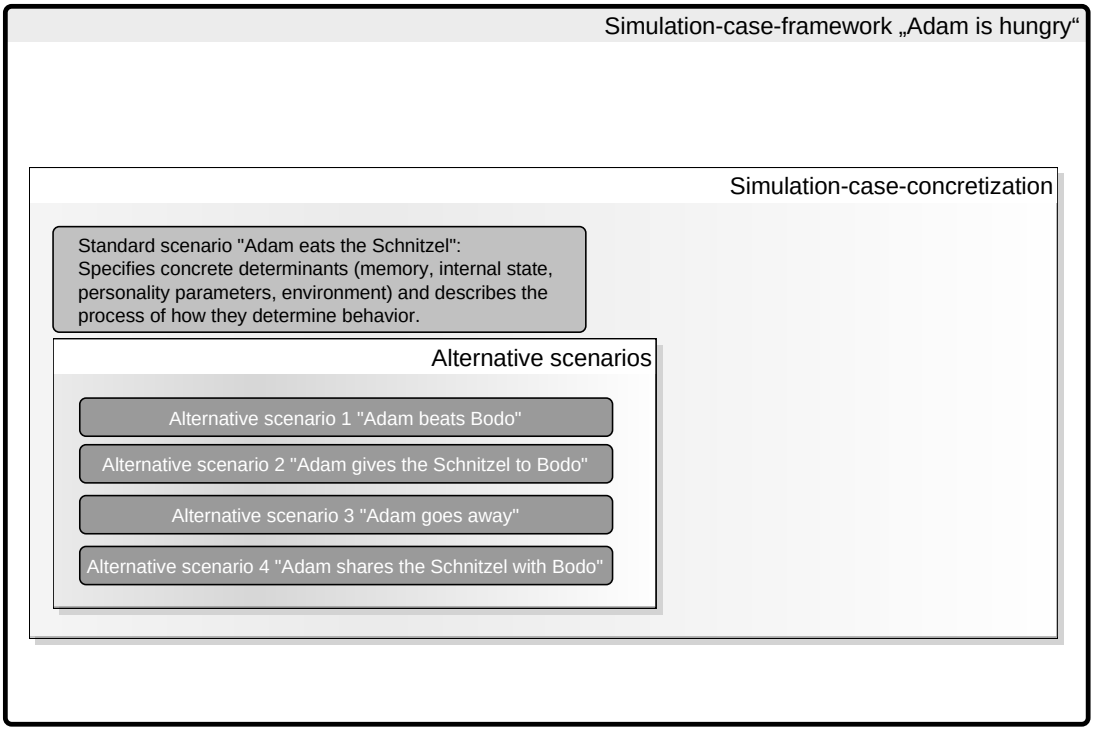


Fig. 2: Overview of a simulation case

In general, the procedure of structuring and transforming the narrative exemplary case to a structured simulation case description is inspired by use-case-based requirements analysis in software engineering. Pre-conditions for the case are represented by the determinants, post-conditions by the description of the agent's final state and the selected actions. Due to the focus on enabling a deterministic description, any change in the system's behavior which results in alternative scenarios must be justified and causally tracked via determinants. This includes describing how these changes in determinants cause change in the system's overall behavior, which supports the development of functions. Typical use-cases are concerned with describing the system's behavior, and requirements for the development of the system that generates this behavior are extracted from them. Due to the scientific nature of simulation-cases, assumptions about how the described behavior is generated are also included. All these aspects enable the use of the simulation case as a shared representation for all involved disciplines. If the simulation of the developed model does not generate the predicted behavior, which of course is normal in a scientific process, we analyze which step to adapt (feedback arrows in Fig. 1) in the next cycle of development and evaluation. This corresponds to an iterative approach to performing science (see chapter 2) in a Laktosian sense.
Overall, such a simulation case structure enables a deterministic and structured version of the exemplary case, fine-grained requirements analysis, and development of a deterministic and causal functional model as well as its evaluation.

3.4 Requirements Analysis

The presented methodology helps to determine requirements in a methodic way. Such requirements state assumptions about the features needed to generate the described behavior. Of course, the behavior description itself represents implicit requirements, since it specifies the desired behavior outcome, but it is only used to analyze the functions required to generate this behavior. Hence we use the behavioral requirements (e.g. perceiving and focusing on a food source) to deduct – with the help of experts – the functional requirements (e.g. perception for the fulfillment of desires, attention mechanisms). The conceptual requirements are analyzed first, i.e. which concepts are required to generate the described

behavior (e.g. motivation based on bodily needs, social norms etc.)? In most cases, the process description (of the inner processes) in the exemplary case specifies the required psychological and sociological concepts. But the transformation into a technical model may trigger new conceptual requirements, which in turn create model requirements to be fulfilled by the implementable agent model (e.g. the interface between body and decision unit, environmental coupling). In developing such a model, implementation requirements may feed back into conceptual and/or model requirements (see feedback possibilities a, b, c in Fig. 1). This is especially true since not only functional requirements but also data requirements are analyzed. For instance, required parameterization of functions may adapt the simulation case's defined data determinants (e.g. the increase of hunger).

3.5 Data and Function Model

Requirements analysis supports finding an algorithmic and knowledge representation of the required function and data model, eventually leading to functions that generate the desired behavior and data that determines it. This step specifies and formalizes the theories employed in the exemplary case in a computerized consistent model. In a first step, an input/output model is used to approach the required functions, i.e. defining the available input and required output; together with the functional requirements, an algorithmic description is developed. In an iterative top-down process, this leads to the development of function modules. The resulting cognitive architecture is termed ARS (Artificial Recognition System) [15]. Since the goal of this paper is to provide an overview of the methodology of development and evaluation, the description of the ARS model is outside of its scope (for an overview see [15]). In principle, however, any cognitive architecture may be used and adapted according to the simulation case's analyzed requirements.

4 Simulation and Evaluation

The resulting model is evaluated by simulating it according to the simulation case scenarios, i.e. using it as a plan to test its predictability. In particular, the simulation is parameterized according to the scenario's data determinants described in the simulation case. The simulation validates whether the assumptions of the simulation case, i.e. the micro-specifications, generate the expected behavior. This can be viewed as testing hypotheses, in particular the hypothesis that the interplay of the developed functions will generate the expected behavior. This includes testing whether the interplay of the various "decision-makers", i.e. the different demands from drives, emotions, environment, reasoning etc., lead to the specified behavior. But we also have to keep in mind that computers only do what we tell them to do [9]. In this regard, we can observe that the agent's behavior is not predetermined directly. For instance, the agent decides what to do based on the valuation of goals. Using a multi-level valuation model to prioritize the agent's goals [13], valuation occurs under different principles and influences. In particular, different functions prioritize different goals in multiple steps. Only at the end of this process, an overall determination of a specific goal's priority is reached. That is, given a certain level of interaction between system components, a person is practically no longer able to deduct and predetermine the system's behavior (cf. complexity).

Validation against the exemplary case's requirements occurs by observing whether the model behaves as expected in the simulation. Additionally, using the same functional model to generate differing behavior (by changing the data according to the specified data determinants) assures that we do not develop a hard-coded behavior model which determines required behavior directly. In fact, the claim that changing specific data determinants leads to different behavior represents a testable hypothesis in itself.

Next we sketch how a change of determinants in the simulation changes the behavior results, exemplified with two scenarios. Simulation of the standard scenario (see Fig. 2) with the specified determinants results in Adam eating the Schnitzel. The corresponding internal state is shown via so called inspectors which visualize data processing (e.g. see Fig. 3 and 4). Thus we can not only validate that the specified data determines the behavior as expected, but also observe if the agent has chosen the behavior according to the assumptions from the exemplary script.

The alternative scenario 1 involves changing sexual drives and Adam's memory, which results in a behavior change that corresponds to Adam's internal state (see Fig. 4). This means that although hungry, Adam decides to beat the other agent based on his valuation and memory. If the agent behaves unexpectedly or the inspectors indicate wrong assumptions, we have to conduct an analysis and a further iteration of the procedure. Possibilities a) and b) in Fig. 1 occur when the inputs from other disciplines (psychoanalysis, neuroscience) are interpreted and transformed incorrectly or implicit requirements emerge during implementation or simulation (implementing a model helps to understand it and

demonstrate contradictions). Possibility c) represents inconsistencies in an underlying theory or between different theories. This feedback thus helps to sharpen theories from other disciplines. Of course it might also be the case that the core assumptions (in a Lakatosian sense) are flawed.

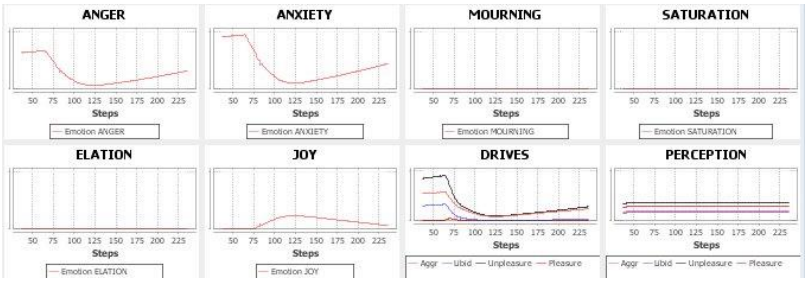


Fig. 3: Adam's emotions and drives in the eating scenario. Eating reduces unpleasure caused by the hunger drive; pleasure is generated as a consequence. Due to Adam's personality, the aggressive part of hunger is satisfied better than the libidinous part. The internal state, together with triggered memories, is represented by emotions.

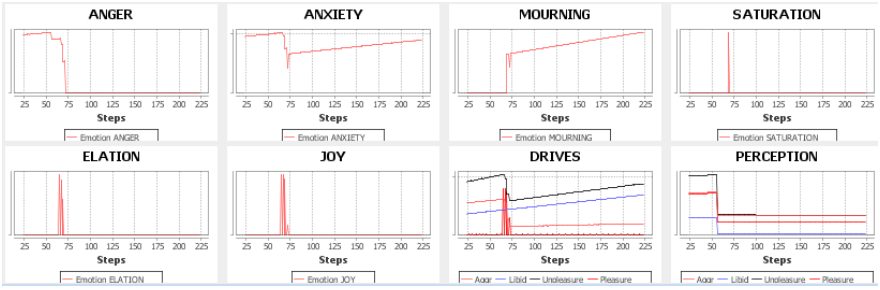


Fig. 4: Changing Adam's memory, which is used to valuate the perceived situation, and increasing his sexual drives results in the decision to beat Bodo. According to Adam's memory, beating brings short but intense pleasure-gain, which leads to the corresponding emotions.

5 Discussion and Conclusion

We have shown that a combination of casuistry, agent-based simulation and requirement engineering is an appropriate methodology for interdisciplinary collaboration to concretize testable assumptions from psychoanalysis and neuroscience. The usage of concrete cases to exemplify research questions pertaining to the development of a functional cognitive architecture with human-like capabilities facilitates interdisciplinary collaboration. In particular, these cases support connecting models and fostering understanding between computer scientists, psychoanalysts and neuroscientists. However, the usage of such an interdisciplinary platform and guidelines poses the need for structuration and determination, in order to be able to develop and evaluate a simulation model. Transforming the exemplary case into a simulation case enables this step and allows for fine-grained requirements analysis.

After finding a functional model that fulfils the simulation case's requirements, it is evaluated in simulations. Evaluation occurs by (1) comparing the desired agent's behavior with the simulated behavior, (2) checking whether the agent's internal state corresponds to the simulation case, and (3) determining whether changing the simulation determinants results in the expected behavioral changes (without changing any functions). Here the structured simulation case is used as a concrete test-plan and makes such detailed evaluation possible. The consideration of different feedback cycles allows iterative development and evaluation. Additionally, it supports interdisciplinary exchange by revealing inconsistencies between assumptions in different disciplines, or within individual disciplines. In this regard, we observed that such methodology forces psychoanalysts and neuroscientists to sharpen their concepts.

The explained evaluation procedure can also be viewed as validating the calibration of the model. Based on this, we can test whether the model generates plausible results (without defining them beforehand in a test plan). Additional evaluation may use collected empirical data. In the case of human decision making, acquiring empirical data about how specific parameters affect decisions is currently impossible, in particular in the case of detailed parameters that represent the inner state (e.g. emotions, memories). To

tackle this restricted accessibility of the human mind we use a case-driven approach in cooperation with specialists in the respective fields: psychoanalysts and neuroscientists use their experience and established evidence to describe, in exemplary cases, how (and why) humans behave. These experience-based hypotheses and the hypotheses on how determinants change the agent's behavior are testable and falsifiable through simulation. In a strict sense, this procedure does not validate the underlying theories of the mind, but does show that they are plausible candidate explanations. In the end, we have shown that our model functions as required in cases that are based on the experience and evidence of specialists. Additionally, validating the determinability of behavior shows the predictability and robustness of the model. All these aspects emphasize that the developed methodology is appropriate to test the plausibility of the developed model. This accountable plausibility is an important step in understanding the human mind's functionality and harnessing its capabilities for technical systems.

References

[1] P. Langley, J.E. Laird, and S. Rogers, Cognitive architectures: Research issues and challenges, *Cognitive Systems Research*, 10:141-160, 2009.

[2] J.R. Anderson, C.S. Carter, J.M. Fincham, Y. Qin, S.M. Ravizza, and M. Rosenberg-Lee, Using fMRI to Test Models of Complex Cognition, *Cognitive Science*, 32: 1323-1348, 2008.

[3] D. Budhitama Subagdja, A. Tan, and G. Ng, Creating Human-like Autonomous Players in Real-time First Person Shooter Computer Games. *Proceedings of the Twenty-First Innovative Applications of Artificial Intelligence Conference*, 2009.

[4] J.R. Anderson, and C. Lebiere, ACT-R: The Newell Test for a theory of cognition. *Behavioral and Brain Sciences*, 26:587-601, 2003.

[5] A. Newell, *Unified Theories of Cognition*, Harvard University Press, 1990.

[6] N.A. Taatgen, Poppering the Newell Test. *Behavioral and Brain Sciences*, 26:621-622, 2003.

[7] R.P. Cooper, The Role of Falsification in the Development of Cognitive Architectures: Insights from a Lakatosian Analysis. *Cognitive Science*, 31:509-533, 2007.

[8] J. Epstein, *Generative Social Science. Studies in Agent-Based Computational Modeling*, Princeton University Press, 2006.

[9] H. Simon, *The Sciences of the Artificial*, Cambridge, MIT Press. 1996.

[10] E.R Smith, and F.R. Conrey, Agent-Based Modeling: A New Approach for Theory Building in Social Psychology, *Personality and Social Psychology Review*, 11:87-104, 2007.

[11] R. Axelrod, Advancing the Art of Simulation in the Social Sciences, *Japanese Journal for Management Information Systems*, 12:1-19, 2003.

[12] P. Davidsson, J. Holmgren, H. Kyhlbäck, D. Mengistu, and M. Persson, Applications of agent based simulation, *Proceedings of the 2006 international conference on Multi-agent-based simulation VII*, 2007.

[13] S. Schaat, K. Doblhammer, D. Dietrich, A Multi-level Model of Motivations and Valuations for Cognitive Agents, *Proceedings of the 6th International Conference on Agents and Artificial Intelligence (ICAART)*, 2014.

[14] A. Damasio, *Looking for Spinoza: Joy, Sorrow, and the Feeling Brain*, Orlando, Harcourt Books, 2003.

[15] D. Dietrich, S. Schaat, D. Bruckner, K. Doblhammer, and G. Fodor, The Current State of Psychoanalytically-Inspired AI. A Holistic and Unitary Model of Human Psychic Processes, *Proceedings of the 39th Annual Conference of the IEEE Industrial Electronics Society*, 2013.

Stratified action negation for dynamic logic

Xin Sun [a] Huimin Dong [b]

[a] *Faculty of Science, Technology and Communication, University of Luxembourg*
[b] *Philosophy and Economics, University of Bayreuth*

Abstract

This paper discusses action negation in dynamic logic. We develop a new action negation called stratified action negation. We classify action by particle, atomic action and calculate its negation step by step. Such calculation makes stratified action negation sensitive to the pre-condition of actions. We axiomatise this logic via combinatory propositional dynamic logic.

Key words: action negation, logic of action, dynamic logic

1 Introduction

Propositional dynamic logic (PDL) is a variant of modal logic and has been developed as a formal system for reasoning about the dynamic behavior of programs. PDL became important for many application areas which could made use of it quite successfully. For instance, among other reasoning about knowledge [4], reasoning about action [3] and deontic logic [6, 9].

Dynamic logic involves not only standard logical connectives ($\vee, \wedge, \neg$), but also action operators (choice: $\cup$, concurrent execution: $\cap$, sequence: ;, iteration: $*$, action negation: $^{-}$). Among all those action operators, action negation is of special interest to logicians [6, 1, 2, 10].

In the dynamic logic literature [5, 7], action negation is usually interpreted as set theoretical complement with respect to the universal relation. Such treatment is not a smart choice when dynamic logic is applied to the deontic setting. Jan Broersen [2] argues that a good notion of action negation for deontic logic should satisfy the following requirement:

- It should have an intuitive interpretation as an action forming operator.
- It should not impose restrictions on the use of other relevant action operator.
- It should have a meaningful interpretation in the normative context.

Presumably the only up to date interpretation of action negation that satisfies the above requirement is Broersen's relativized action negation [1, 2]. However, in this paper we argue that in addition to Broersen's requirement, the ideal notion of action negation should further satisfy the following requirement:

$^{(*)}$For two different actions α and β, in general we should **not** have $\alpha \cup \overline{\alpha} = \beta \cup \overline{\beta}$.

For an illustration of $^{(*)}$, consider the following scenario:

- suppose Hamlet receives the following authorization:

  (1) "You are permitted either to be or not to be."

- and James Bond receives the following authorization:

  (2) "You are permitted either to kill or not to kill. "

Abstract away the factor of agents, (1) offers the agent a free choice between to live and to be dead while (2) offers the agent the license to kill or not. These two permissions convey very different information and should be distinguished. That is, the permission of $\alpha \cup \overline{\alpha}$ and $\beta \cup \overline{\beta}$ should be different, which gives us a reason to distinguish $\alpha \cup \overline{\alpha}$ and $\beta \cup \overline{\beta}$.

Both the traditional action negation and Broersen's relativized action negation do not satisfy this requirement. In this paper we develop a new interpretation of action negation such that it is intuitively acceptable and satisfies the above requirement.

2 Dynamic logic

Let $\mathbb{P}$ be a countable set of propositional letters and $\mathbb{A}$ a countable set of symbols of action generators. The language of dynamic logic can be defined by the following BNF:

Definition 1 (Language of dynamic logic) *For $a \in \mathbb{A}$ and $p \in \mathbb{P}$,*

- $\alpha := \mathbf{1} \mid a \mid \alpha \cup \alpha \mid \alpha \cap \alpha \mid \alpha; \alpha \mid \alpha^* \mid \phi? \mid \overline{\alpha}$
- $\phi := p \mid \top \mid \neg\phi \mid \phi \wedge \phi \mid [\alpha]\phi$

Here symbols of the form α are action terms and ϕ are formulas. We use $\langle\alpha\rangle\varphi$ as an abbreviation of $\neg[\alpha]\neg\varphi$. Formulas are interpreted by relational models, which is defined as follows.

Definition 2 (Relational model) *A relational model $M = (W, R^{\mathbb{A}}, V)$ is a triple:*

- *W is a nonempty set of possible states.*
- *$R^{\mathbb{A}} : \mathbb{A} \to 2^{W \times W}$ is an action interpretation function, assigning a binary relation over W to each action generator $a \in \mathbb{A}$.*
- *$V : \mathbb{P} \to 2^W$ is the valuation function for propositional letters.*

The action interpretation function $R^{\mathbb{A}}$ is extended to a new function R to interpret arbitrary actions as follows:

- $R(\mathbf{1}) = W \times W$,
- $R(a) = R^{\mathbb{A}}(a)$ for $a \in \mathbb{A}$,
- $R(\alpha \cup \beta) = R(\alpha) \cup R(\beta)$,
- $R(\alpha \cap \beta) = R(\alpha) \cap R(\beta)$,
- $R(\alpha; \beta) = R(\alpha) \circ R(\beta)$,
- $R(\phi?) = \{(w, w) \mid w \in V(\varphi)\}$,
- $R(\alpha^*) = (R(\alpha))^*$.

Here $\circ$ is the composition operator for relations and * is the reflexive transitive closure operator of relations. We leave the case for $R(\overline{\alpha})$ to the next section because that is the theme of this paper. With the function R in hand, we can define the semantics for formulas of dynamic logic use relational model as following:

Definition 3 (Semantics of dynamic logic) *Let $M = (W, R^{\mathbb{A}}, V)$ be a relational model. Let $w \in W$.*

- *$M, w \models p$ iff $w \in V(p)$*
- *$M, w \models \neg\phi$ iff not $M, w \models \phi$*
- *$M, w \models \phi \wedge \phi$ iff $M, w \models \phi$ and $M, w \models \phi$*
- *$M, w \models [\alpha]\phi$ iff for all v, if $(w, v) \in R(\alpha)$ then $M, v \models \phi$*

3 Action Negation

Broersen [2] suggests the negation of action α should be different from α, an alternative of α, and can be considered as the act of refraining from α. We accept such intuition. In this section we review the known treatment for dynamic logic on action negation, then we define a new, and better, treatment to implement such intuition.

3.1 Action negation in the literature

The traditional interpretation of action negation [5] is to let $R(\overline{\alpha}) = (W \times W) - R(\alpha)$, *i.e.* the set theoretical complement with respect to the universal relation. Broersen [1, 2] argues that the universal relation is not the ideal background for complement when dynamic logic is applied to deontic logic. The reason is, given such interpretation, $\alpha \cup \overline{\alpha}$ represents the universal relation. Therefore the modality $[\alpha \cup \overline{\alpha}]$ has universal power and can reach every state, including those states which are not reachable by any action. Apparently our arbitrary action, such as "write a paper or not write a paper", don't have such strong power. Therefore the traditional interpretation of action negation is not appropriate for deontic logic, or even the logic of action.

To avoid such predicament, Broersen [2] restricts the universal relation such that those worlds which are unreachable by any action are out of concern. With this motivation Broersen replaces the universal relation $W \times W$ in the interpretation of $R(\overline{\alpha})$ by relations like $\bigcup_{\alpha \in \mathbb{A}} R(\alpha)$, $(\bigcup_{\alpha \in \mathbb{A}} R(\alpha))^+$, $(\bigcup_{\alpha \in \mathbb{A}} R(\alpha))^*$ etc.

Another proposal for action negation is developed by Wansing [10]. Wansing suggests that an operator "$-$" for action is an operation of action negation as long as it satisfies the following: $R(-(\alpha;\beta)) = R(-\alpha;-\beta)$, $R(-(\alpha^*)) = R((-\alpha)^*)$ and $R(-(\varphi?)) = R((\neg\varphi)?)$. Broersen [2] points out that this approach is not ideal because it it might be too liberal. For instance we can define for all action α, $\alpha = -\alpha$, assuming we have no operator '?' in our language. Such definition still satisfies $R(-(\alpha;\beta)) = R(-\alpha;-\beta)$ and $R(-(\alpha^*)) = R((-\alpha)^*)$, but contradicts to our intuition.

Broersen's approach is more natural than its traditional counterpart in the deontic setting. But there is a drawback, as we have already mentioned in the introductory section: the action $\alpha \cup \overline{\alpha}$ and $\beta \cup \overline{\beta}$ are identical in Broersen's approach. More generally, any interpretation of action negation validate the following is problematic: for all action α and β, $R(\alpha \cup \overline{\alpha}) = R(\beta \cup \overline{\beta})$. In the following section we develop a new interpretation of action negation such that$R(\alpha \cup \overline{\alpha}) \neq R(\beta \cup \overline{\beta})$ in general.

3.2 A new approach: stratified action negation

We first make a classification about actions. Since actions are interpreted by relations and the simplest relation is a set that contains an ordered pair of states, we can naturally call an action α **particle** if $R(\alpha)$ contains exactly one ordered pair. For a particle action α, we call the first component of the ordered pair in $R(\alpha)$ the pre-state of α. Formally, if $R(\alpha) = \{(s_1, s_2)\}$, then $pre(\alpha) = \{s_1\}$. And we call the second component the post-state of α, formally $post(\alpha) = \{s_2\}$. Intuitively, a particle action is simply a travel from one state to another.

Based on particle action, we build **atomic action** as a union of particle actions which share the same pre-state. For instance, for two particle actions α_1 and α_2 with $R(\alpha_1) = \{(s_1, s_2)\}$, $R(\alpha_2) = \{(s_1, s_3)\}$, the action α_3 such that $R(\alpha_3) = \{(s_1, s_2), (s_1, s_3)\}$ is an atomic action. See Figure 1 .

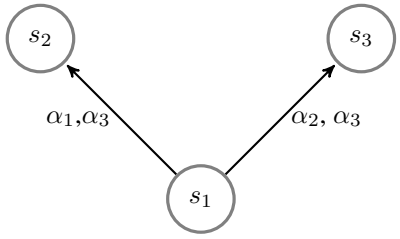


Figure 1: atomic action

For an atomic action α, its pre-state is the same as its consisting particle actions. The post condition of α is the union of the post-state of its consisting particle actions. Therefore $post(\alpha_3) = \{s_2, s_3\}$. Intuitively, an atomic action is a nondeterministic travel from a specific state to other states.

For an arbitrary action α, we defined its pre-state as the union of the pre-state of its consisting particle actions. Formally, $pre(\alpha) = \{s \in W | (s,t) \in R(\alpha)\}$.

Now we have a classification of actions and the pre/post state of an action has been defined. It is the time to grasp what the negation of an action is. For an atomic action α, say $R(\alpha) = \{(t,t),(t,u)\}$ and $W = \{t,u,v\}$, we tend to define $R(\overline{\alpha}) = \{(t,v)\}$. See Figure 2.

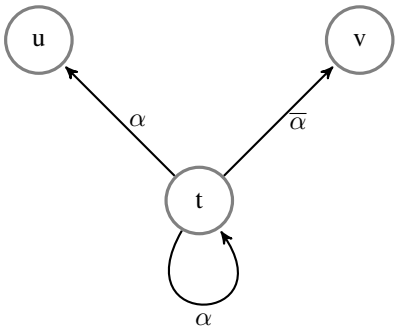


Figure 2: atomic action negation

The intuition is, we understand α as a plan of travel from t to either u or t itself, then the negation of α can be understood as "go to those states other than the states α goes". Suppose t, u, v represent Thailand, UK and Vietnam respectively, the action α in Figure 2 is understood as "stay in Thailand, or go to UK from Thailand". The negation of α is "go to Vietnam from Thailand". More formally, we define $R(\overline{\alpha}) = pre(\alpha) \times (W - post(\alpha))$ for an atomic action α.

For an arbitrary action α, we calculate $R(\overline{\alpha})$ via the following steps:

1. We first decompose α to atomic actions $\alpha_1, \ldots, \alpha_n$ such that $R(\alpha) = R(\alpha_1) \cup \ldots \cup R(\alpha_n)$ and for every $i \neq j$, $pre(\alpha_i) \neq pre(\alpha_j)$. It can be verified that such decomposition is unique and each α_i is a maximal atomic sub-action of α in the sense that for every atomic action β, if $R(\beta) \subseteq R(\alpha)$ then there exist a unique α_i in the decomposition such that $R(\beta) \subseteq R(\alpha_i)$

2. For each $i \in \{1, \ldots, n\}$, we then calculate $R(\overline{\alpha_i})$. Since α_i is an atomic action, we have $R(\overline{\alpha_i}) = pre(\alpha_i) \times (W - post(\alpha_i))$.

3. Finally we take the union of these $R(\overline{\alpha_i})$ to form $R(\overline{\alpha}) = R(\overline{\alpha_1}) \cup \ldots \cup R(\overline{\alpha_n})$.

Since we calculate the action negation by decomposing complex actions to simple actions, we call such an approach **stratified action negation**. For an example, if $W = \{u,v,w\}$, $R(\alpha) = \{(w,w),(w,u),(v,w)\}$, then $R(\overline{\alpha}) = \{(w,v),(v,v),(v,u)\}$. See Figure 3.

It will then be nice to have a uniform formal representation of the above procedure. To achieve this we give the following definition:

Definition 4 (Stratified action negation) *For an arbitrary action α, we define $R(\overline{\alpha}) = Pre(\alpha) \times W - R(\alpha)$, if $R(\alpha) \neq \emptyset$. Otherwise we let $R(\overline{\alpha}) = W \times W$.*

The above definition makes the negation of an action be sensitive to its pre-conditions. We believe such treatments is intuitively acceptable. According to Definition 4, we have for two action α and β, as long as $pre(\alpha) \neq pre(\beta)$, we have $R(\alpha \cup \overline{\alpha}) \neq R(\beta \cup \overline{\beta})$.

3.3 Axiomatization via combinatory propositional dynamic logic

Our stratified action negation can be expressed in language of combinatory propositional dynamic logic (CPDL) [8]. In Passy and Tinchev [8], CPDL is an extension of PDL with nomials to name every state.

Let $\mathbb{P}, \Sigma$ be countable sets of propositional letters and nominals respectively, $\mathbb{A}$ a countable set of symbols of action generators. We require $\mathbb{P}, \Sigma$ and $\mathbb{A}$ to be disjoint. The language of CPDL can be defined by the following BNF:

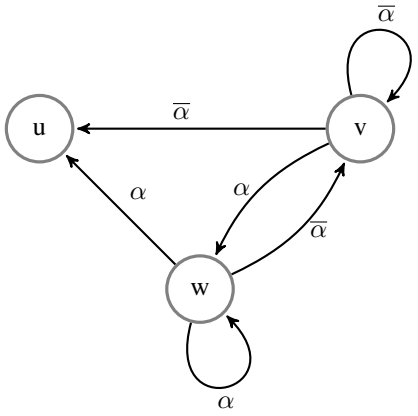


Figure 3: arbitrary action negation

Definition 5 (Language of CPDL [8]) *For $a \in \mathbb{A}$, $i \in \Sigma$ and $p \in \mathbb{P}$,*

- $\alpha := \mathbf{1} \mid a \mid \alpha \cup \alpha \mid \alpha \cap \alpha \mid \alpha;\alpha \mid \alpha^* \mid \phi? \mid \widetilde{\alpha} \mid \alpha^{-1}$
- $\phi := p \mid i \mid \top \mid \neg\phi \mid \phi \wedge \phi \mid \langle\alpha\rangle\phi \mid \alpha = \alpha \mid \alpha \subset \alpha$

Here $\widetilde{\alpha}$ refers to traditional action negation in the sense that $R(\widetilde{\alpha}) = W \times W - R(\alpha)$. α^{-1} refers to the converse of α in the sense that $R(\alpha^{-1}) = \{(t,s) \mid (s,t) \in R(\alpha)\}$.

Definition 6 (Relational model with nominal [8]) *A relational model with nominal is a quadruple $\mathbb{M} = (W, R^{\mathbb{A}}, \chi, V)$:*

- *W is a nonempty set of possible states.*
- *$R^{\mathbb{A}} : \mathbb{A} \to 2^{W \times W}$ is an action interpretation function, assigning a binary relation over W to each action generator $a \in \mathbb{A}$.*
- *$\chi : \Sigma \to W$ is a surjective function.*
- *$V : \mathbb{P} \to 2^W$ is the valuation function for propositional letters.*

The action interpretation function $R^{\mathbb{A}}$ is extend to a new function R to interpret arbitrary actions as follows:

- $R(\mathbf{1}) = W \times W$,
- $R(a) = R^{\mathbb{A}}(a)$ for $a \in \mathbb{A}$,
- $R(\alpha \cup \beta) = R(\alpha) \cup R(\beta)$,
- $R(\alpha \cap \beta) = R(\alpha) \cap R(\beta)$,
- $R(\alpha;\beta) = R(\alpha) \circ R(\beta)$,
- $R(\alpha^*) = (R(\alpha))^*$,
- $R(\phi?) = \{(w,w) \mid w \in V(\varphi)\}$,
- $R(\widetilde{\alpha}) = W \times W - R(\alpha)$,
- $R(\alpha^{-1}) = R(\alpha)^{-1}$.

Definition 7 (Semantics of CPDL [8]) *Let $M = (W, R^{\mathbb{A}}, \chi, V)$ be a relational model with nominal. Let $w \in W$,*

- $M, w \models i$ *iff* $w = \chi(i)$,
- $M, w \models \alpha = \beta$ *iff for all* v, $(w, v) \in R(\alpha)$ *iff* $(w, v) \in R(\beta)$,
- $M, w \models \alpha \subset \beta$ *iff for all* v, *if* $(w, v) \in R(\alpha)$ *then* $(w, v) \in R(\beta)$.

Other cases are the same as PDL.

The following is the proof-system of CPDL [8].

1. Axiom Schemes:
   - All proposition tautologies.
   - $\langle \mathbf{1} \rangle i$
   - $\langle \mathbf{1} \rangle (i \wedge \varphi) \rightarrow [\mathbf{1}](i \rightarrow \varphi)$
   - $\varphi \rightarrow \langle \mathbf{1} \rangle \varphi$
   - $\langle \mathbf{1} \rangle \langle \mathbf{1} \rangle \varphi \rightarrow \langle \mathbf{1} \rangle \varphi$
   - $\varphi \rightarrow [\mathbf{1}] \langle \mathbf{1} \rangle \varphi$
   - $\langle \alpha \rangle \varphi \rightarrow \langle \mathbf{1} \rangle \varphi$
   - $\langle \alpha ; \beta \rangle \varphi \leftrightarrow \langle \alpha \rangle \langle \beta \rangle \varphi$
   - $\langle \alpha \cup \beta \rangle i \leftrightarrow \langle \alpha \rangle i \vee \langle \beta \rangle i$
   - $\langle \alpha \cap \beta \rangle i \leftrightarrow \langle \alpha \rangle i \wedge \langle \beta \rangle i$
   - $\langle \widetilde{\alpha} \rangle i \leftrightarrow [\alpha] \neg i$
   - $\alpha \subset \beta \leftrightarrow [\alpha \cap \widetilde{\beta}] \bot$
   - $\langle \mathbf{1} \rangle (i \wedge \langle \alpha^{-1} \rangle j) \leftrightarrow \langle 1 \rangle (j \wedge \langle \alpha \rangle i)$
   - $\langle \varphi ? \rangle \psi \leftrightarrow \varphi \wedge \psi$
   - $\langle \alpha^* \rangle \varphi \leftrightarrow \varphi \vee \langle \alpha \rangle \langle \alpha^* \rangle \varphi$
   - $[\alpha](\varphi \rightarrow \psi) \rightarrow ([\alpha]\varphi \rightarrow [\alpha]\psi)$
2. Rules:
   - If $\vdash [\alpha] \neg i$ for all $i \in \Sigma$, then $\vdash [\alpha] \bot$
   - If $\vdash [\beta][\alpha^i]\varphi$ for all $i \in \mathbb{N}$, then $\vdash [\beta][\alpha^*]\varphi$
   - If $\vdash \varphi$, then $\vdash [\mathbf{1}]\varphi$
   - If $\vdash \varphi$ and $\vdash \varphi \rightarrow \psi$, then $\vdash \psi$

It is proved by Passy and Tinchev [8] that the above proof-system is sound and complete with the class of relational models with nominal. Our stratified action negation can be expressed in the language of CPDL. To achieve this, first note that the notion of pre-condition and stratified action negation can both semantically be defined using notions of CPDL.

Proposition 1 *Let* M *be an arbitrary relational model with nominal,*

1. $w \in pre(\alpha)$ *iff* $M, w \models \langle \alpha \rangle \top$,
2. $M, w \models [\overline{\alpha}]\varphi$ *iff* $M, w \models \langle \alpha \rangle \top \rightarrow [\widetilde{\alpha}]\varphi$.

Proof:

1. (left-to-right) Assume $w \in pre(\alpha)$, then there exist some $v \in W$ such that $(w, v) \in R(\alpha)$. Therefore $M, w \models \langle \alpha \rangle \top$.

   (right-to-left) Assume $M, w \models \langle \alpha \rangle \top$, then there exist some $v \in W$ such that $(w, v) \in R(\alpha)$ and $M, v \models \top$. Therefore $w \in pre(\alpha)$.

2. (left-to-right) Assume $M, w \models [\overline{\alpha}]\varphi$, then $M, v \models \varphi$ for all v such that $(w, v) \in pre(\alpha) \times W - R(\alpha)$. Note that $pre(\alpha) \times W - R(\alpha) = pre(\alpha) \times W \cap R(\widetilde{\alpha})$. Hence for all v such that $(w, v) \in pre(\alpha) \times W \cap R(\widetilde{\alpha})$, $M, v \models \varphi$. Now if $M, w \models \langle\alpha\rangle\top$, then by item 1 we know $w \in pre(\alpha)$.

   Let u be an arbitrary state such that $(w, u) \in R(\widetilde{\alpha})$. We know that $(w, u) \in pre(\alpha) \times W$ because $w \in pre(\alpha)$. Therefore $(w, u) \in pre(\alpha) \times W \cap R(\widetilde{\alpha})$. Hence $M, u \models \varphi$, $M, w \models [\widetilde{\alpha}]\varphi$.

(right-to-left) Assume $M, w \models \langle\alpha\rangle\top \rightarrow [\widetilde{\alpha}]\varphi$. Let v be an arbitrary state such that $(w, v) \in pre(\alpha) \times W - R(\alpha)$. Then $(w, v) \in pre(\alpha) \times W \cap R(\widetilde{\alpha})$. So we have $pre(\alpha) \neq \emptyset$ and $M, w \models \langle\alpha\rangle\top$. Therefore $M, w \models [\widetilde{\alpha}]\varphi$. From $(w, v) \in R(\widetilde{\alpha})$ we deduce $M, v \models \varphi$. Therefore $M, w \models [\overline{\alpha}]\varphi$. □

Therefore we can syntactically let $[\overline{\alpha}]\varphi$ be an abbreviation of $\langle\alpha\rangle\top \rightarrow [\widetilde{\alpha}]\varphi$. Then the axiomatization of CPDL is also the axiomatization of our stratified action negation.

4 Conclusion

This paper discusses action negation in dynamic logic. We develop a new action negation called stratified action negation. We classify action by particle, atomic action and calculate its negation step by step. Such calculation makes stratified action negation sensitive to the pre-condition of actions. We axiomatise this logic via combinatory propositional dynamic logic.

Acknowledgment

The first author thanks Dov Gabbay for his inspiring lectures given in the University of Luxembourg.

References

[1] Jan Broersen. Relativized action negation for dynamic logics. In P. Balbiani, N-Y. Suzuki, F. Wolter, and M. Zakharyaschev, editors, *Advances in Modal Logic*, volume 4, pages 51–70, 2003.

[2] Jan Broersen. Action negation and alternative reductions dynamic deontic logics. *Journal of applied logic*, 2004.

[3] Giuseppe De Giacomo and Maurizio Lenzerini. Pdl-based framework for reasoning about actions. In Marco Gori and Giovanni Soda, editors, *Topics in Artificial Intelligence*, volume 992 of *Lecture Notes in Computer Science*, pages 103–114. Springer Berlin Heidelberg, 1995.

[4] Ronald Fagin, Joseph Y. Halpern, Yoram Moses, and Moshe Y. Vardi. *Reasoning about knowledge*. MIT Press, 1995.

[5] D. Harel, D. Kozen, and J. Tiuryn. *Dynamic logic*. The MIT Press, 2000.

[6] John Jule Meyer. A different approach to deontic logic: deontic logic viewed as a variant of dynamic logic. *Notre Dame Journal of Formal Logic*, pages 109–136, 1988.

[7] Ewa Orlowska. Dynamic logic with program specifications and its relational proof system. *Journal of Applied Non-Classical Logics*, 3(2):147–171, 1993.

[8] Solomon Passy and Tinko Tinchev. Pdl with data constants. *Information Processing Letters*, 20(1):35–41, 1985.

[9] Ron van der Meyden. The dynamic logic of permission. *Journal of Logic and Computation*, 6:465–479, 1996.

[10] Heinrich Wansing. On the negation of action types: Constructive concurrent pdl. *Dresden Preprints in theoretical philosophy and philosophical logic*, 2004.

How do pessimistic agents save miners? A STIT based approach

Xin Sun [a] Zohreh Baniasadi [a] Shuwen Zhou [b]

[a] *Faculty of Science, Technology and Communication, University of Luxembourg*
[b] *School of computer Science Engineering, University of New South Wales*

Abstract

This paper develops a new STIT based deontic logic, pessimistic utilitarian deontic logic, capable of analyzing the miners puzzle. The key idea of the semantics of this logic is: one set of possible worlds is better than another set of possible worlds iff the worst world in the first set is better than the worst world in the second. This semantics give right predictions in the miners scenario meanwhile blocks the derivation to contradiction.

1 Introduction

This paper develops a new STIT based deontic logic, referring it as pessimistic utilitarian deontic logic, capable of analyzing a recently popular puzzle in deontic logic, the miners puzzle [9]. The miners puzzle goes like this:

> Ten miners are trapped either in shaft `A` or in shaft `B`, but we do not know which one. Water threatens to flood the shafts. We only have enough sandbags to block one shaft but not both. If one shaft is blocked, all of the water will go into the other shaft, killing every miner if they are inside. If we block neither shaft, both will be partially flooded, killing one miner.

Lacking any information about the miners' exact whereabouts, it seems acceptable to say that:

(1) We ought to block neither shaft.

However, we also accept that

(2) If the miners are in shaft `A`, we ought to block shaft `A`.

(3) If the miners are in shaft `B`, we ought to block shaft `B`.

But we also know that

(4) Either the miners are in shaft `A` or they are in shaft `B`.

And (2)-(4) seem to entail

(5) Either we ought to block shaft `A` or we ought to block shaft `B`.

Which contradicts (1).

It is stated by Willer [14] that any adequate semantics of dyadic deontic modality must offer a solution to the miners puzzle. The existing STIT-based deontic logic [7, 10, 13] does not offer a satisfying analysis to this puzzle: although the deduction from (2)-(4) to (5) is blocked by the dyadic deontic operator defined in Sun [13], both Horty [7] and Sun [13] are unable to predict (1). In this paper our motivation is to develop a new STIT-based deontic logic which is capable of blocking the deduction from (2)-(4) to (5) and it is able to predict (1)-(4).

In STIT-based deontic logic, agents make choices. Every choice of every agent is represented by a set of possible worlds. A preference relation over possible worlds is given as primitive. This preference relation is then lifted to the relation of preference over sets of worlds. A choice is better than another iff the set of worlds representing the first choice is better than the set of worlds representing the second. A proposition φ is obligatory (we ought to see to it that φ) iff it is ensured by every best choice, i.e., it is true in every world of every best choice.

Therefore the interpretation of deontic modality is based on best choices, which can only be defined on top of preference over sets of worlds, which is defined by lifting from the preference over worlds. There is no standard way of lifting preference. Lang and van der Torre [12] discuss the following three ways of lifting:

- **strong lifting** For two sets of worlds W_1 and W_2, W_1 is strongly better than W_2 iff $\forall w \in W_1$, $\forall v \in W_2$, w is better than v. That is, the worst world in W_1 is better than the best world in W_2.
- **optimistic lifting** W_1 is optimistically better than W_2 iff $\exists w \in W_1, \forall v \in W_2$, w is better than v. That is, the best world in W_1 is better than the best world in W_2.
- **pessimistic lifting** W_1 pessimistically better than W_2 iff $\exists v \in W_2, \forall w \in W_1$, w is better than v. That is, the worst world in W_1 is better than the worst world in W_2.

In Horty [7], Kooi and Tamminga [10] and Sun [13] the strong lifting is adopted. Applying the strong lifting to the miners scenario, all the three choices *block_neither*, *block_*A and *block_*B are best. "we ought to block neither" is then not true in the miners scenario in the logic of Horty [7], Kooi and Tamminga [10] and Sun [13].

In this paper we use pessimistic lifting instead of strong lifting. There is a single best choice *block_neither* according to pessimistic lifting. Therefore "we ought to block neither" is true. It is further proved that both (2) and (3) are true while the deduction from (2)-(4) to (5) is not valid. Therefore our logic offers a satisfying solution to the miners puzzle.

The structure of this paper is as follows: in Section 2 we review the existing solutions to the miners puzzle. Then in Section 3 we review the existing STIT-based deontic logic. In Section 4 we develop the pessimistic utilitarian deontic logic and offer a viable solution to the miners puzzle. Section 5 is our conclusion and future work.

2 Solutions of the miners paradox

Several authors have provided different solutions to solve the miners puzzle. Among them, we summarize the following approaches:

Kolodny and MacFarlane [9] give a detailed discussion of various escape routes. Then they conclude that the only possible solution to the puzzle is to invalidate the argument from (2) to (5). To do this, Kolodny and MacFarlane state we have three choices: rejecting modus ponens (MP), rejecting disjunction introduction ($\vee$I), rejecting disjunction elimination ($\vee$E). Among these three Kolodny and MacFarlane further demonstrate that the only wise choice is to reject MP.

Willer [14] developed a fourth option to invalidate the argument form (2) to (5): falsify the monotonicity. In his solution the modus ponens can be preserved (there are very good reasons to do so) and we are unable to derive the inconsistency.

Charlow [5] proposes a comprehensive solution which requires rethinking the relationship between relevant information (what we know) and practical rankings of possibilities and actions (what to do).

Cariani et al [3] argue that the traditional Kratzer's semantics [11] of deontic conditionals is not capable of solving the puzzle. They propose to extend the standard Kratzer's account by adding a parameter representing a "decision problem" to solve the puzzle. Roughly, a decision problem contains a representation of action and a decision rule to select best action. Cariani et al [3] use a partition of all possible worlds to represent actions, and the decision rule they used to select action is essentially the same as the MaxiMin principle–the decision theoretic rule that requires agents to evaluate actions in terms of their worst conceivable outcome and choose the 'least bad" one among them. Such treatment shares some similarity with a special case of our logic to be in Section 4. In our logic every agent's actions are also represented by a partition of all worlds. And we use pessimistic lifting to compare actions, which is the same as MaxiMin.

Carr [4] argues that the proposal of Cariani et al is still problematic. To develop a satisfying semantics, Carr uses three parameters to define deontic modality: an informational parameter, a value parameter and a decision rule parameter. According to Carr's proposal, (1) to (3) are all correct predictions and no contradiction arise within her framework.

Gabbay et al [6] offers a solution to the miners puzzle using idea from intuitionistic logic. In their logic "or" is interpreted in an intuitionistic favour. Then the reasoning from statement (2) to (5) is blocked.

3 STIT-based deontic logic

In this section we review STIT-based deontic logic. Following Horty [7], we call such logic utilitarian deontic logic (UDL).

3.1 Language

The language of the UDL is built from a finite set $Agent$ of agents and a countable set P of propositional letters. We will use p, q as variables for atomic propositions in P, and use G, with $G \subseteq Agent$, as a group of agents. The utilitarian deontic language L is given by the following Backus-Naur Form:

$$\phi ::= p \mid \neg\phi \mid \phi \wedge \phi \mid [G]\phi \mid \bigcirc_G \phi \mid \bigcirc_G(\phi/\phi)$$

Intuitively, $[G]\phi$ is read as "group G sees to it that ϕ". $\bigcirc_G \phi$ is read as "G ought to see to it that ϕ". $\bigcirc_G(\phi/\psi)$ is read as "G ought to see to it that ϕ under the condition ψ".

3.2 Semantics

The semantics of utilitarian deontic logic is based on the utilitarian frames, which is a simplification of STIT frame of Horty [7].

Definition 1 (Utilitarian frame) *A $utilitarian\ frame$ is a tuple $\langle W, A, Choice, \leq\rangle$, where W is a nonempty set of possible worlds, A is a finite set of agents, $Choice$ is a choice function, and $\leq$, represents the preference of the group A, is a reflexive and transitive relation on W.*

The choice function $Choice$ is a function from the power set of A to the power set of the power set of W, i.e. $Choice : \wp(A) \mapsto \wp(\wp(W))$. $Choice$ is built from the individual choice function $IndChoice$: $A \mapsto \wp(\wp(W))$. The $IndChoice$ must satisfy the following three conditions:

(1) for each $i \in A$ it holds that $IndChoice(i)$ is a partition of W;

(2) let $A = \{1, ..., n\}$, for every $x_1 \in IndChoice(1), \ldots, x_n \in IndChoice(n)$, $x_1 \cap \ldots \cap x_n \neq \emptyset$;

We call a function s: $A \mapsto \wp(W)$ a selection function if for each $i \in A$, $s(i) \in IndChoice(i)$. Let $Selection$ be the set of all selection functions, for any $G \subseteq A$, if $G \neq \emptyset$, we define $Choice(G) = \{\bigcap_{i\in G} s(i) : s \in Selection\}$. If $G = \emptyset$, we define $Choice(G) = \{W\}$.

Here our utilitarian frame simplifies STIT frame in the sense that we restrict STIT frame to a single moment, therefore the concept of history in STIT frame is omitted.

Definition 2 (preferences over sets of worlds via strong lifting [13]) *Let $X, Y \subseteq W$ be two sets of worlds from a utilitarian frame. Then $X \preceq^s Y$ (Y is $weakly\ preferred$ to X) if and only if*

(1) for each $w \in X$, for each $w' \in Y$, $w \leq w'$ and

(2) there exists some $v \in X$, some $v' \in Y$, $v \leq v'$.

$X \prec^s Y$ (Y is $strongly\ preferred$ to X) if and only if $X \preceq^s Y$ and it is not the case that $Y \preceq^s X$.

Definition 3 (dominance relation [7]) *Let F be a utilitarian frame. Let $G \subseteq A$ and $K, K' \in Choice(G)$. Then*

$$K \preceq^s_G K' \quad \textit{iff} \quad \textit{for all } S \in Choice(A - G),\ K \cap S \preceq^s K' \cap S.$$

$K \preceq^s_G K'$ is read as "K' weakly dominates K". From a decision theoretical perspective, $K \preceq^s_G K'$ means that no matter how other agents act, the outcome of choosing K' is no worse than that of choosing K. We use $K \prec^s_G K'$ as an abbreviation of $K \preceq^s_G K'$ but $K' \preceq^s_G K$ does not hold. If $K \prec^s_G K'$, we then say K' strongly dominate K.

Definition 4 (restricted choice sets [7]) *Let G be groups of agents from a utilitarian frame and X a set of worlds in the frame. Then*

$$Choice(G/X) = \{K : K \in Choice(G) \text{ and } K \cap X \neq \emptyset\}$$

Intuitively, $Choice(G/X)$ is the collection of those choices of group G that are consistent with condition X.

We now review conditional dominance relation over agent's choice. The intuition is: to compare whether the agent's choice K is dominated by K' under the condition X, we only need to consider other agents' choices which are consistent with the condition X and at least one of K and K'.

Definition 5 (conditional dominance [13]) *Let G be groups of agents from a utilitarian frame and X a set of worlds in the frame. Let K, $K' \in Choice(G/X)$. Then*

$$K \preceq^s_{G/X} K' \quad \textit{iff} \quad \textit{for all } S \in Choice((A-G)/(X \cap (K \cup K'))),\ K \cap X \cap S \preceq^s K' \cap X \cap S.$$

$K \preceq^s_{G/X} K'$ is read as "K' weakly dominates K under the condition of X". And we will use $K \prec^s_{G/X} K'$, read as "K' strongly dominates K under the condition of X", to express $K \preceq^s_{G/X} K'$ and it is not the case that $K' \preceq^s_{G/X} K$.

Definition 6 (Optimal and conditional optimal [7]) *Let G be a group of agents from a utilitarian frame,*

- $Optimal^s_G$ = $\{K \in Choice(G)$: *there is no* $K' \in Choice(G)$ *such that* $K \prec^s_G K'\}$.
- $Optimal^s_{G/X}$ = $\{K \in Choice(G/X)$: *there's no* $K' \in Choice(G/X)$ *such that* $K \prec^s_{G/X} K'\}$.

As in traditional modal logic, a model is a frame plus a valuation.

Definition 7 (utilitarian model) *A utilitarian model M is an ordered pair (F, V) where F is a utilitarian frame and V a valuation that assigns to each atomic proposition $p \in P$ a set of worlds $V(p) \subseteq W$.*

In the semantic of UDL, the optimal choices and conditional optimal choices are used to interpret the deontic operators.

Definition 8 (truth conditions) *Let $M = (\boldsymbol{F}, V)$ be a utilitarian model. Let $w \in W$ and let $\varphi, \psi \in L$. Then*

(1) $M, w \models p$	*iff*	$w \in$*V(p);*
(2) $M, w \models \neg\varphi$	*iff*	*it is not the case that* $M, w \models \varphi$*;*
(3) $M, w \models \varphi \wedge \psi$	*iff*	$M, w \models \varphi$ *and* $M, w \models \psi$*;*
(4) $M, w \models \bigcirc_G \varphi$	*iff*	$K \subseteq \|\|\varphi\|\|$ *for each* $K \in Optimal^s_G$*;*
(5) $M, w \models \bigcirc_G(\varphi/\psi)$	*iff*	$K \subseteq \|\|\varphi\|\|$ *for each* $K \in Optimal^s_{G/\psi}$*.*
(6) $M, w \models [G]\phi$	*iff*	$M, w' \models \phi$ *for all* $w' \in W$ *such that there is* $K \in Choice(G)$, $\{w, w'\} \subseteq K$*;*

Here $\|\varphi\| = \{w \in W : M, w \models \varphi\}$.

We say φ is true in the world w of a utilitarian model M if $M, w \models \varphi$. Just like in the standard modal logic (for instance, Blackburn [1]), we introduce the concept of validity as follows: a formula φ is valid ($\models \varphi$) iff it is true at every world of every utilitarian model.

4 Pessimistic utilitarian deontic logic

For pessimistic utilitarian deontic logic, instead of strong lifting, we using pessimistic lifting.

Definition 9 (preferences over sets of worlds via pessimistic lifting) *Let $X, Y \subseteq W$ be two sets of worlds from a utilitarian frame. Then $X \preceq^p Y$ if and only if there exists $w \in X$, such that for all $w' \in Y$, $w \leq w'$. $X \prec^p Y$ if and only if $X \preceq^p Y$ and it is not the case that $Y \preceq^p X$.*

Proposition 1 *Let X and Y be sets of worlds from a utilitarian frame. Then:*

1. *If $X \preceq^p Y$ and $Y \preceq^p Z$, then $X \preceq^p Z$.*
2. *If $X \preceq^p Y$ and $Y \prec^p Z$, then $X \prec^p Z$.*
3. *If $X \prec^p Y$ and $Y \preceq^p Z$, then $X \prec^p Z$.*

Proof: Here we prove the first two items, the third case is similar.

1. Assume $X \preceq^p Y$ and $Y \preceq^p Z$, then ther exist $u \in X$ such that for all $v \in Y$, $u \leq v$. There exist $v' \in Y$ such that for all $w \in Z$, $v' \leq w$. Then we have $u \leq v'$. For an arbitrary $w' \in Z$, we have $v' \leq w'$. Therefore $u \leq w'$.
2. Assume $X \preceq^p Y$ and $Y \prec^p Z$. From item *1.* we know $X \preceq^p Z$. To prove it is not the case that $Z \preceq^p X$, we assume otherwise. Then from $Z \preceq^p X$ and $X \preceq^p Y$ we deduce $Z \preceq^p Y$, contradicts to $Y \prec^p Z$. □

Proposition 1 states that the relation of preference over sets of worlds via pessimistic lifting is transitive. It is worth knowing that this proposition is crucial. Only with transitivity, we can properly define the concept of dominance and optimality.

The definition of dominance ($\preceq^p_G$), conditional dominance ($\preceq^p_{G/X}$), optimal ($Optimal^p_G$) and conditional optimal ($Optimal^p_{G/X}$) in the pessimistic setting are obtained by simply changing $\leq^s$ to $\leq^p$ of their counterpart of defined via strong lifting.

Proposition 2 *Let G be a group of agents from a utilitarian frame, and let K, K', $K'' \in Choice(G)$. Then:*

1. *If $K \preceq^p_G K'$ and $K' \preceq^p_G K''$, then $K \preceq^p_G K''$.*
2. *If $K \preceq^p_G K'$ and $K' \prec^p_G K''$, then $K \prec^p_G K''$.*
3. *If $K \prec^p_G K'$ and $K' \preceq^p_G K''$, then $K \prec^p_G K''$.*

Proof: Here we prove the first two cases. The third case is similar.

1. Assume $K \preceq^p_G K'$ and $K' \preceq^p_G K''$, then for all $S \in Choice(A - G)$, $K \cap S \preceq^p K' \cap S$, $K' \cap S \preceq^p K'' \cap S$. Then by Proposition 1 we have $K \cap S \preceq^p K'' \cap S$. Hence $K \preceq^p_G K''$.
2. Similar to the proof of item 2 of Proposition 1. □

Proposition 3 *Let G be a group of agents from a utilitarian frame and X a set of worlds in the frame. Let K, K', $K'' \in Choice(G/X)$. Then the following holds,*

1. *If $K \preceq^p_{G/X} K'$ and $K' \preceq^p_{G/X} K''$, then $K \preceq^p_{G/X} K''$.*
2. *If $K \preceq^p_{G/X} K'$ and $K' \prec^p_{G/X} K''$, then $K \prec^p_{G/X} K''$.*
3. *If $K \prec^p_{G/X} K'$ and $K' \preceq^p_{G/X} K''$, then $K \prec^p_{G/X} K''$.*

Proof: See Proposition 12 of Sun [13]. □

We add $\bigcirc^p_G \phi$ and $\bigcirc^p_G(\phi/\psi)$ to our language L to represent "from the pessimistic perspective, G ought to see to it that ϕ" and "from the pessimistic perspective, G ought to see to it that ϕ in the condition ψ" respectively. The truth condition for $\bigcirc^p_G \phi$ and $\bigcirc^p_G(\phi/\psi)$ are defined as follows:

Definition 10 (truth conditions) *Let M be a utilitarian model and $w \in W$.*

$M, w \models \bigcirc^p_G \phi$ *iff* $K \subseteq ||\phi||$ *for each* $K \in Optimal^p_G$;
$M, w \models \bigcirc^p_G(\phi/\psi)$ *iff* $K \subseteq ||\phi||$ *for each* $K \in Optimal^p_{G/\psi}$.

4.1 Application: an analysis of the miners puzzle

Now we return to the miners scenario. The miners scenario is described formally by a utilitarian model as $Miners = (W, Agent, Choice, \leq, V)$, where $W = \{w_1, \ldots, w_6\}$, $Choice(G) = \{\{w_1, w_2\}, \{w_3, w_4\}, \{w_5, w_6\}\}$, $Choice(Agent - G) = \{W\}$, $w_3 \approx w_6 \leq w_1 \approx w_2 \leq w_4 \approx w_5$, $V(in\_A) = \{w_1, w_3, w_5\}$,$V(in\_B) = \{w_2, w_4, w_6\}$, $V(block\_A) = \{w_5, w_6\}$, $V(block\_B) = \{w_3, w_4\}$,$V(block\_neither) = \{w_1, w_2\}$. We visualize the miners scenario by the following figure:

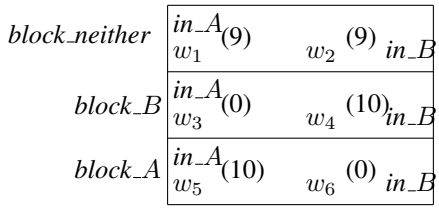


Figure 4.1: $W = \{w_1, \ldots, w_6\}$, $w_3 \approx w_6 \leq w_1 \approx w_2 \leq w_4 \approx w_5$.

According to the pessimistic semantics, *block_neither* is the only optimal choice. So we can draw the prediction that "we ought to block neither" $i.e.\ Miners, w_1 \vDash \bigcirc^p_G(block\_neither)$. Moreover, given the condition of miners being in A, *block_A* becomes the only conditional optimal choice. Hence we have "if the miners are in A, then we ought to block A", $i.e.\ Miners, w_1 \vDash \bigcirc^p_G(block\_A/in\_A)$. The case for miners being in B are similar. Although we have both "if the miners are in A, then we ought to block A" and "if the miners are in B, then we ought to block B", by Proposition 4 below we can avoid the prediction that "we ought to block either A or B". Hence no contradiction arise. Therefore our logic gives right prediction meanwhile avoids contradictions. It therefore offers a viable solution to the miners puzzle.

Proposition 4 $\nvdash \bigcirc^p_G(p/q) \wedge \bigcirc^p_G(p/r) \rightarrow \bigcirc^p_G(p/(q \vee r))$.

Proof: $Miners = (W, Agent, Choice, \leq, V)$ introduced in the beginning of this section serves as a counter example. Just consider in_A as q, in_B as r, $block\_A \vee block\_B$ as p. □

Compared to those approaches reviewed in Section 2, our STIT based approach has stronger expressive power. We have agents and action modality in our language. This gives considerable expressive power already. Just like Horty [7] and Broersen [2], our framework can easily be extended to involve temporal modality, which further increase the expressive power.

5 Conclusion and future work

This paper develops a new STIT based deontic logic, pessimistic utilitarian deontic logic, capable of analyzing the miners puzzle. The key idea of the semantics of this logic is: one set of worlds is better than another set of worlds iff the worst world in the first set is better than the worst world in the second. This semantics gives right predictions in the miners scenario meanwhile blocks the derivation of contradiction.

Concerning future works, an axiomatisation of pessimistic utilitarian deontic logic is worthy investigating. A second potential extension is to use other STIT operators, for example the deliberative STIT [8] and X-STIT [2].

References

[1] Patric Blackburn, Maarten de Rijke, and Yde Venema. *Modal Logic*. Cambridge University Press, Cambridge, 2001.

[2] Jan Broersen. Deontic epistemic *stit* logic distinguishing modes of mens rea. *Journal of Applied Logic*, 9(2):127 – 152, 2011.

[3] Fabrizio Cariani, Magdalena Kaufmann, and Stefan Kaufmann. Deliberative modality under epistemic uncertainty. *Linguistics and Philosophy*, 36(3):225–259, 2013.

[4] Jennifer Carr. Deontic modals without decision theory. *Proceedings of Sinn und Bedeutung*, 17:167–182, 2012.

[5] Nate Charlow. What we know and what to do. *Synthese*, 190(12):2291–2323, 2013.

[6] Dov M. Gabbay, Livio Robaldo, Xin Sun, Leendert van der Torre, and Zohreh Baniasadi. Toward a linguistic interpretation of deontic paradoxes - beth-reichenbach semantics approach for a new analysis of the miners scenario. In Fabrizio Cariani, Davide Grossi, Joke Meheus, and Xavier Parent, editors, *Deontic Logic and Normative Systems - 12th International Conference, DEON 2014, Ghent, Belgium, July 12-15, 2014. Proceedings*, volume 8554 of *Lecture Notes in Computer Science*, pages 108–123. Springer, 2014.

[7] John Horty. *Agency and Deontic Logic*. Oxford University Press, New York, 2001.

[8] John Horty and Nuel Belnap. The deliberative stit: a study of action, omission, ability, and obligation. *Journal of Philosophical Logic*, 24:583–644, 1995.

[9] Niko Kolodny and John MacFarlane. Iffs and oughts. *Journal of Philosophy*, 107(3):115–143, 2010.

[10] Barteld Kooi and Allard Tamminga. Moral conflicts between groups of agents. *Journal of Philosophical Logic*, 37:1–21, 2008.

[11] Angelike Kratzer. The notional category of modality. In H. J. Eikmeyer and H. Rieser, editors, *Words, worlds, and Contexts: New Approaches in World Semantics*. Berlin: de Gruyter, 1981.

[12] Jérôme Lang and Leendert van der Torre. From belief change to preference change. In *Proceedings of Eighteenth European Conference on Artificial Intelligence (ECAI2008)*, pages 351–355, 2008.

[13] Xin Sun. Conditional ought, a game theoretical perspective. In J. Lang H. van Ditmarsch and S. Ju, editors, *Logic, Rationality, and Interaction: Proceedings of the Thire International Workshop*, pages 356–369, Guangzhou, China, 2011.

[14] Malte Willer. A remark on iffy oughts. *Journal of Philosophy*, 109(7):449461, 2012.

Dynamic Lateral Stability for an Energy Efficient Gait

Zhenglong Sun [a] Nico Roos [a]

[a] *Department of Knowledge Engineering, Maastricht University*
P.O. Box 616, 6200 MD Maastricht, The Netherlands

Abstract

This paper presents an energy efficient dynamically stable gait for a Nao humanoid robot. In previous work we identified a dynamically stable and energy efficient gait in the sagittal or walking direction of a Nao robot. This gait proved to be more energy efficient than the standard gait, provided by the manufacturer. Dynamic stability in the lateral direction was not addressed. Lateral stability was handled by full stiffness of the joint in lateral direction. In this paper we report on adding dynamic lateral stability. We do not yet incorporate feedback of sensors. This implies that the gait is only suited for flat horizontal surfaces that some lateral joint stiffness is needed in the implementation on the Nao.

1 Introduction

The gait of humans is often assumed to be the most energy-efficient way of walking [2]. Srinivasan and Ruina [10] confirm this hypothesis using a simple model in which the human is a point mass with straight legs that can change in length during a step. Their results show that the dynamically stable inverted pendulum walk is the most energy-efficient gait [7]. In previous work we demonstrated this also holds for humanoid robots such as a Nao, despite differences with humans. For instance, humans do not need to bend the knee of the stance leg while walking, because they can push off using the foot and the calf muscle. A humanoid robot such as a Nao, cannot push off using its foot. Instead it must provide the energy for maintaining a gait by bending and stretching its knee joint. Experiments with a Nao showed that the torque of knee joint as a result of bending the knee joint is the main source of energy consumption of the Nao during walking. A dynamically stable gait in the sagittal direction that minimize the knee bending proved to be the most energy-efficient gait.

We extend the dynamically stable gait in the sagittal direction with dynamic stability in the lateral direction. Simulations with an inverted pendulum model for the Nao with dynamic lateral stability showed that adding dynamic lateral stability does not change the dynamically stable gait in sagittal direction. We determined the requirements for the lateral stability and we adapted our gait controller incorporating dynamic lateral stability. The adapted gait controller has 8 control parameters, for which we learned the optimal values for a Nao through policy gradient algorithm. Figure 1 shows a high-level outline of the approach.

In the paper, we first briefly describe our previous work [11, 12] about a walking pattern generation which identifies a most energy-efficient gait without considering lateral stability (Section 2). The main contribution of this paper is to propose a method for lateral stability improvement (Section 3). The results are used to create a gait controller (Section 4) which is fine-tuned for a Nao robot using a *webots* simulator (Section 5). The gait is evaluated on a real Nao robot (Section 6). The last section concludes the paper.

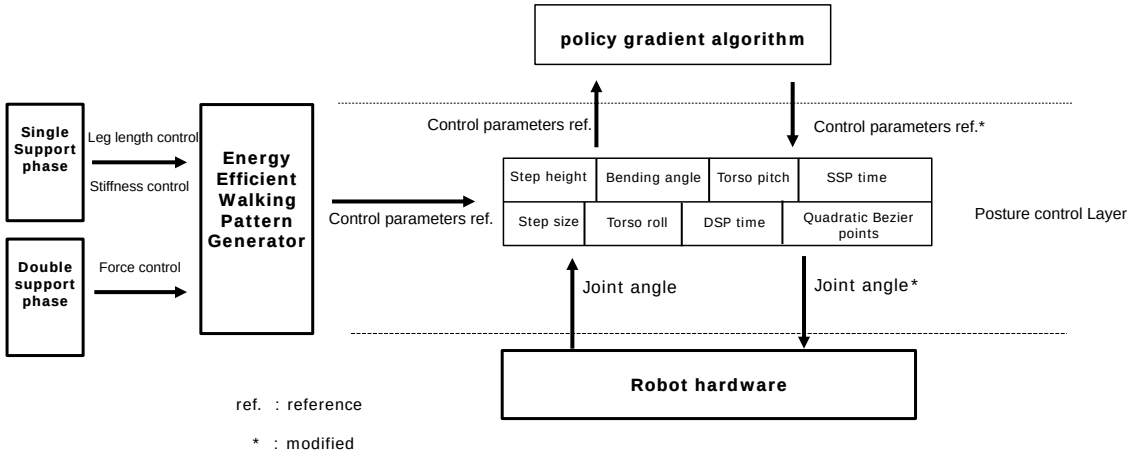


Figure 1: Outline of the proposed approach, showing the details about work flow

2 Walking Pattern Generation

The gait of humans and of humanoid (bipedal) robots is a repeating pattern consisting of two phases; a single support phase (SSP) where the body is supported by only one leg and a double support phase (DSP) where the body is supported by both legs [5]. In the DSP the weight of the body is shifted from one leg to the other. The DSP is crucial for the lateral stability but is sometimes ignored when analysing the gait. However, since it is impossible to implement a gait on a Nao without a double support phase, we must consider it in our model. We will start presenting a model without a DSP and subsequently extend the model with a DSP.

2.1 Single Support Phase

To analyse the energy consumption, we developed an Inverted Pendulum model with telescopic legs [11, 12]. This model, which is based on the work of Srinivasan and Ruina [10], allows the length of the support leg to vary during a step. A leg-length policy $\delta : [-\frac{\pi}{2}, \frac{\pi}{2}] \rightarrow [0, 1]$ determines how much the stance leg will be shortened as function of the angle β between stance leg with vertical axis. The shortening of the stance leg is realized by bending the knee joint. Experiment with a Nao robot showed the the torque on the knee joint is the main factor determining the energy consumption during walking and that the energy needed to stretch the stance leg during a step can be ignored. Using simulations in Matlab, an optimal leg-length policy that minimize the energy consumption, was determined [11, 12]. The optimal leg-length policy shows that the SSP starts with a slightly bended stance leg which is subsequently stretched. After stretching the stance leg remains stretched till the end of the step.

2.2 Double Support Phase

We extended the model described in the previous subsection with double support phase [4]. The length of the SSP during a step will be a parameter of our controller.

We need a way to describe the influence of the swing leg on the mass in DSP. This can not be done by just simply applying a leg-length policy for the swing leg in the double support phase. Given the step size s, the leg-length policy $\delta(\beta)$ of the stance leg and angle β, the length leg of the swing leg is fixed. Prescribing the length of the swing leg by a policy, creates a rigid triangle in which the mass m can no longer move freely. We therefore choose to let the mass m move freely given the leg-length policy of the stance leg and use a force policy for the swing leg in the double support phase. So the length of the swing leg is determined by the leg-length policy of the stance leg, the step size and the angle of the stance leg, but the force that swing leg executes on the mass is determined by the force policy of the swing leg. This force may influence the sagittal speed of the mass m.

After adding the double support phase and the force policy for the swing leg during the double support phase, we re-run our Matlab simulation. The result of this simulation showed that the optimal

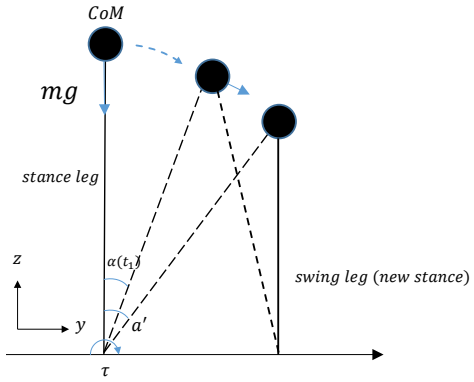


Figure 2: The lateral plane during double support phase

leg-length policy did not change and the optimal force policy is to put no force on the swing leg till it becomes the new stance leg. We also evaluated the effects of different force polices on leg-length policy. The Matlab experiments showed that the shape of the leg-length policy does not change. The robot still starts with a slightly bended leg which is subsequently stretched and remains stretched till the end of the step. During the DSP, the stance leg is always stretched.

In the double support phase the robot has to shift its weight from the stance leg to the swing leg. In order to keep balance in lateral direction at the end of the double support phase, the robot must put force on the swing leg to stop the lateral movement in time. Adding this observation to our model and to our gait controller for the Nao robot is the main contribution of this paper.

3 Lateral Control

To improve the stability of the walking pattern generation described previously, we exploit lateral controller to regulate the CoM lateral movement and velocity during double support phase. Our idea is to use the force generated by the swing leg and and upper body tilt to regulate lateral component of CoM velocity. Since this work focus on the lateral component of the walking motion, and our experiments with different leg-length polices showed the stance leg is always stretched during the DSP, we restrict the equations to the lateral plane. Missura and Behnke [8] confirmed that sagittal and lateral controllers can be modeled independently.

In order to introduce the equations describe the movement of CoM in lateral plane, we first define a variable $\alpha(t)$ which is the angle between the stance leg and the vertical axis. When the total force resulting from gravity and inertia generates rotation around the contact point between the sole of stance leg and ground, the angle $\alpha(t)$ varies from 0 to α', as illustrated in Figure 2.

We assume that during the single support phase, the robot is perfectly balanced in the lateral direction. Therefore, at the beginning of DSP, the stance leg is vertical to the ground ($\alpha(0) = 0$), and in the lateral plane there is no torque making the CoM rotate around the sole of the stance leg. In order to generate the torque τ rotating the CoM from $\alpha(0)$ to α', we manipulate the upper body to bend slightly inwards at angle ω. The bending ω disrupts the balance enabling gravity to create a torque $\tau > 0$. We manipulate the force generated by the swing leg to control the rotation of the CoM with a non-zero angular velocity $\dot{\alpha}$ and to stop at the position ($\alpha(t) = \alpha'$) where the robot can put its whole body weight on the new stance leg and keep it stable. The problem is to control the torque τ appropriately. Our method to mitigate the problem is controlling the force generated by the swing leg by means of a force policy. Compared to the height of CoM, the step size is small (less than 5% of CoM height). Therefore we assume that in the DSP, the length of stance leg can be considered as fixed in the lateral plane. That is, on the stance leg, we ignore the effect of the CoM moving forward in the sagittal plane. Given this assumption, we analyse the forces on the CoM in the lateral plane. We break down the gravity into two components consisting of a radial force F_r along the stance leg and F_p that is perpendicular to the stance leg (see Figure 3.a). We manipulate the force policy $\gamma(\alpha)$ of the swing leg to control the force F_q generated by the swing leg on the CoM. We also break down F_q into two components: F_b which directs opposite of F_r and another force F'_p which is opposite to F_p (see Figure 3.b). Therefore, in the radial

direction, there is the combine radial force $F_n = F_r + F_b$. Since the length of stance leg l is fixed, the radial force F_n has no effect on the movement of the CoM. However, the combine force $F_t = F_p + F'_p$ generates a torque τ around the contact point between the sole of stance leg and ground. The torque τ rotates the CoM, shifting the weight from stance leg to the swing leg.

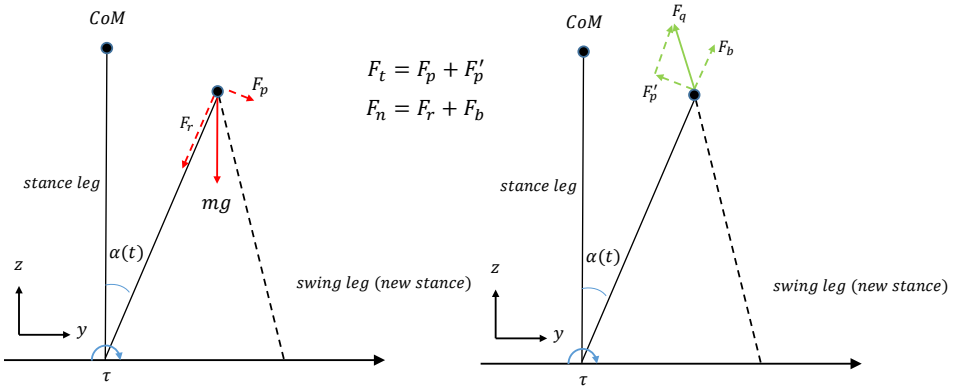


Figure 3: (a) Force on stance leg in lateral plane and (b) Forces on swing leg in lateral plane

At the beginning of DSP, ideally, the force policy $\gamma(\alpha)$ imposes no force on swing leg. Therefore the torque τ generated by the slightly inward bending ω of upper body is needed to start the lateral movement of the CoM. As $\alpha(t)$ increases, the force policy controls the force F_q to gradually decrease the $\dot{\alpha}(t)$, and stops the CoM movement when $\alpha(t) = \alpha'$. The force F_q causes the force F'_p which de-accelerates the movement of the CoM. When $\alpha(t) = \alpha'$, the force $F_t = 0$ and the torque $\tau = F_t \cdot l = 0$, and therefore the CoM will stop rotating. We generate a force policy by regulating the knee stiffness of the swing leg, as a function of the angle $\alpha(t)$. The shape of the force policy is determined by means of Quadratic Bezier curves, as illustrated in Figure 4. The Quadratic Bezier curves is defined by 3 points in the interval of the DSP. The start point and the end point are fixed, so we start with no force generated by the swing leg and stop with the full weight of the robot on the swing leg, after which it becomes the new stance leg. We assume a smooth transition between these two points which is determined by the middle point ζ of the Quadratic Bezier curves. So we have to determine the optimal point ζ.

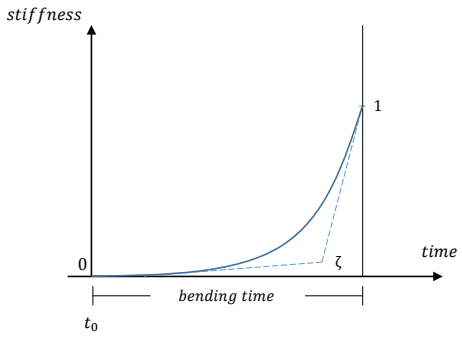


Figure 4: Stiffness over time by Quadratic Bezier Curves

To summarize, the controller manipulates the upper body to bend slightly inwards at angle ω to trigger the CoM movement. At first there is no stiffness on swing leg, therefore the magnitude of F'_p is 0. Consequently, F_t leads to an acceleration of the angular velocity of CoM around the sole of the stance leg, which make the knee joint of the swing leg start to bend. Next, the stiffness of the keen joint on swing leg is increased in order to stop the rotation when the CoM reaches its end position ($\alpha(t) = \alpha'$).

4 Optimizing Gait Parameter for Nao

This section describes the learning of the optimal control parameter of a dynamic gait for a Nao.

4.1 Gait Parameters

This section presents the parameters of a gait that realizes the leg-length policy determined by the experiments described in the previous section. Based on the results of the simulation experiments, we identify 8 parameters (the new parameters to this work are the Quadratic Bezier points and the Torso Roll inclination, see Section 3) that are essential in controlling a dynamic gait:

- *Step Size* (θ_1): Defines the how long Nao can move in a singe step (sagittal).
- *Step Height* (θ_2): Defines the maximum distance between ground and lifting feet. A high step height will require higher speed of the swing leg and may cause horizontal instability. A low step height increases the possibility of tripping and limits the step size.
- *Knee Bending* (θ_3): Defines the maximum bending of the swing leg at the beginning of the double support phase. This parameter determines the sagittal velocity and the energy cost.
- *SSP Time* (θ_4): Defines how long the single support phase lasts. This parameter determines the sagittal walking velocity.
- *DSP Time* (θ_5): Defines how long the double support phase lasts. This parameter determines the duration of the swing leg (the next stance leg) compression to θ_3 in the double support phase.
- *Torso Pitch Inclination* (θ_6): Defines the maximum angle that torso leans in sagittal direction. If positive, it will move the center of mass (CoM) in sagittal direction. If it is set not appropriate, a fall will occur.
- *Quadratic Bezier points* (ζ): Defines the magnitude of middle points in Quadratic Bezier Curves which determines the ground reaction force on swing leg (introduced in Section 3).
- *Torso Roll Inclination* (ω): Defines the maximum angle that torso leans in lateral direction. If positive, it will move the center of mass (CoM) towards the swing leg in frontal view. If it is set not appropriate, instability will occur (introduced in Section 3).

Table 1: Trajectory Parameters in Sagittal Plane

Description of joint motion	q
step size	θ_1
swing hip pitch	$p_1(\theta_1, \theta_2, \theta_4, \theta_6)$
swing knee pitch	$p_2(\theta_1, \theta_2, \theta_4, \theta_6)$
swing ankle pitch	$-p_1 - p_2$
stance hip pitch	$p_3(\theta_1, \theta_2, \theta_3, \theta_5, \theta_6)$
stance knee pitch	$p_4(\theta_1, \theta_2, \theta_3, \theta_5, \theta_6)$
stance ankle pitch	$-p_2 - p_4$

Table 1 shows all the parameters of the trajectory for walk movement. The walk posture q is determined by joints value, step size, acceleration and so on. The value of hip pitch, knee pitch and ankle pitch are functions p_n of parameter sets.

4.2 Policy Gradient Algorithm

After investigating several policy search algorithms [3, 9], we chose to use a policy gradient method presented by Kohl and Stone [6] to optimize the Nao's gait. This method is among the Finite-different methods and quite straightforward to understand. In the control law optimization experiment [9], the finite-different methods turned out to be less efficient than other prominent general approaches and converge to local optimal in motor planning experiments, Nevertheless, if we cannot differentiate the polices w.r.t the control parameters, the finite different methods becomes the only option applicable [9]. In this method, the objective function Q is a function to be optimized for the energy cost and stability.

The policy gradient method starts with an initial parameter vector $\pi = \theta_1, ..., \theta_N$ and estimates the partial derivative of the objective function Q with respect to each parameter. This is done by evaluating

t randomly generated policies $R_1, ...R_t$ near π, such that each $R_i = \theta_1 + \delta_1, ..., \theta_N + \delta_N$ and δ_j is randomly chosen to be either $-\epsilon$, 0, $+\epsilon$, where ϵ is a small fixed value relative to θ. After evaluating each policy R_i on the objective function *Q*, each dimension of every R_i is grouped into one of the three categories to estimate an average gradient for each dimension:

$$R_i \in \begin{cases} S_{+\epsilon,n}, & \text{if the } n\text{th parameter of } R_i \text{ is } \theta_n + \epsilon_n \\ S_{+0,n}, & \text{if the } n\text{th parameter of } R_i \text{ is } \theta_n + 0 \\ S_{-\epsilon,n}, & \text{if the } n\text{th parameter of } R_i \text{ is } \theta_n - \epsilon_n \end{cases}$$

We calculate average score $Avg_{-\epsilon,n}$, $Avg_{+0,n}$ and $Avg_{+\epsilon,n}$ for $S_{-\epsilon,n}$, $S_{+0,n}$ and $S_{+\epsilon,n}$ respectively.

- $Avg_{-\epsilon,n}$ average score for all R_i that have a negative perturbation in dimension *n*
- $Avg_{+0,n}$ average score for all R_i that have a zero perturbation in dimension *n*
- $Avg_{+\epsilon,n}$ average score for all R_i that have a positive perturbation in dimension *n*

These three average values estimate the benefit of altering the nth parameter by $+\epsilon_n$, 0, $-\epsilon_n$. An adjustment vector A of size n is calculated where

$$A_n = \begin{cases} 0, & \text{if } Avg_{+0,n} \geq Avg_{+\epsilon,n} \text{ and} \\ & Avg_{+0,n} \geq Avg_{-\epsilon,n} \\ Avg_{+\epsilon,n} - Avg_{-\epsilon,n}, & \text{otherwise} \end{cases}$$

In order to generate a gait that is energy efficient and stable, we adopt an objective function based on

Algorithm 1 Pseudo-code of Policy Gradient Algorithm

$\pi \leftarrow InitialPolicy$
while $!done$ **do**
  $R_1, R_2, ...R_t = t\ random\ perturbations\ of\ \pi$
  $evaluate(R_1, R_2, ...R_t)$
  for $n = 1\ to$ **N do**
    $Avg_{+\epsilon,n}$
    $Avg_{+0,n}$
    $Avg_{-\epsilon,n}$
    if $Avg_{+0,n} > Avg_{+\epsilon,n}$ and $Avg_{+0,n} > Avg_{-\epsilon,n}$ **then**
      $A_n \leftarrow 0$
    else
      $A_n \leftarrow Avg_{+\epsilon,n} - Avg_{-\epsilon,n}$
    end if
  end for
  $A \leftarrow \frac{A}{|A|} * \eta$
  $\pi \leftarrow \pi + A$
end while

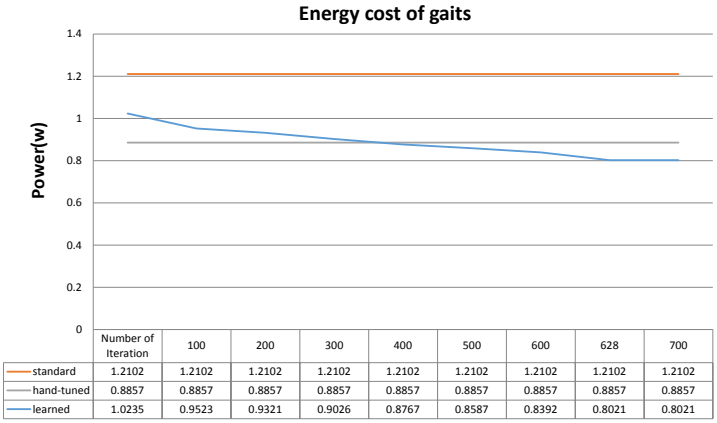
the energy cost and the stability. The energy cost is expressed by the normalized current M_c, and the stability by normalized standard deviation of the three accelerometers M_a

$$Q = 1 - (w_c M_c + w_a M_a) \tag{1}$$

The components of the objective function are weighted by w_c and w_a respectively to optimize for desirable goal. These weights are constrained so that the sum of the weights are equal to one. In this experiment, we set w_c = 0.25, w_a = 0.75.

4.3 Learning optimal parameters in simulator

To generate the optimal gait parameters and test the gait's performance, two separate experiments were conducted. Firstly, we upload local modules into simulator *Webots* to run the policy gradient algorithm.



Number of Iteration	100	200	300	400	500	600	628	700
standard	1.2102	1.2102	1.2102	1.2102	1.2102	1.2102	1.2102	1.2102	1.2102
hand-tuned	0.8857	0.8857	0.8857	0.8857	0.8857	0.8857	0.8857	0.8857	0.8857
learned	1.0235	0.9523	0.9321	0.9026	0.8767	0.8587	0.8392	0.8021	0.8021

Figure 5: Comparison of Energy Cost of Three Gaits

We used a relatively elementary hand-tune gait as a starting policy for the policy gradient algorithm described in Section V. Though a bad starting policy may lead to a simulation failure, we did not deliberately optimize the starting policy. The performance of the task is measured by monitoring the knee torque and electric current. Falling or "toddle" is penalized. We found a most energy-efficient and dynamically stable gait after 628 iterations. Subsequent evaluations showed no further improvement. The learning algorithm produce a set parameters of stable gait which is more energy efficient and faster than the standard walk of Nao. The parameters of the gait and ϵ value are given in Table 2.

Table 2: Initial Parameters and Best Learned Parameters

Parameter	Initial Value	ϵ	Learned Value
step size	6(cm)		6(cm)
step height	3(cm)	0.02	3.24(cm)
knee bending	15 (degree)	0.1	13.8 (degree)
SSP time	300(ms)	25	225(ms)
DSP time	300(ms)	25	375(ms)
torso pitch inclination	10 (degree)	0.1	7.5 (degree)
torso roll inclination	10 (degree)	0.1	6.8
Quadratic Bezier points	(0.5*DSP time, 0.5)	0.1	(0.9*DSP time, 0.2)

5 Simulation and Real World Evaluation

To validate our approach, we perform a real world experiment with the Nao humanoid robot, which has 25 degrees of freedom. We validate the result of the learned parameters by sending them to a Nao robot and command it to walk a constant distance. We compared the energy consumption of learned gait with the energy consumption of the standard gait of the Nao and our initial hand-tuned gait. The step size was set to 6 cm. Figure 5 shows the energy consumption of the three gaits. We see that the learned gait results in a power reduction of 33.7% of standard gait and a reduction of 9.4% of hand-tune gait. The accompanying video material[1] shows the Nao robot walking on flat ground with our proposed gait controller.

6 Conclusion

In this paper, we presented a framework to generate energy efficient dynamic human-like walk for a Nao humanoid robot. Based on previous work, we proposed a simple lateral control method for a gait with dynamic lateral stability. We optimize the control policy for a Nao humanoid robot and evaluated the

[1] https://project.dke.maastrichtuniversity.nl/robotlab/wp-content/uploads/naowalk.mp4

result on the real Nao. The result shows the gait we proposed is more energy efficient and dynamically stable than the standard Nao gait.

References

[1] JJ Alcaraz-Jiménez, D Herrero-Pérez, and H Martınez-Barberá. A closed-loop dribbling gait for the standard platform league. In *Workshop on Humanoid Soccer Robots of the IEEE-RAS Int. Conf. on Humanoid Robots (Humanoids)*, 2011.

[2] Steven H Collins and Andy Ruina. A bipedal walking robot with efficient and human-like gait. In *ICRA*, pages 1983–1988. IEEE, 2005.

[3] Marc Peter Deisenroth, Roberto Calandra, André Seyfarth, and Jan Peters. Toward fast policy search for learning legged locomotion. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pages 1787–1792. IEEE, 2012.

[4] Joel A DeLisa. *Gait analysis in the science of rehabilitation*, volume 2. Diane Publishing, 1998.

[5] Arnaud Hamon, Yannick Aoustin, et al. Optimal walking gait with double support, simple support and impact for a bipedal robot equipped of four-bar knees. *IMSD 2012*, 2012.

[6] Nate Kohl and Peter Stone. Machine learning for fast quadrupedal locomotion. In *AAAI*, volume 4, pages 611–616, 2004.

[7] Arthur D Kuo. The six determinants of gait and the inverted pendulum analogy: A dynamic walking perspective. *Human movement science*, 26(4):617–656, 2007.

[8] Marcell Missura and Sven Behnke. Lateral capture steps for bipedal walking. In *Humanoid Robots (Humanoids), 2011 11th IEEE-RAS International Conference on*, pages 401–408. IEEE, 2011.

[9] Jan Peters and Stefan Schaal. Policy gradient methods for robotics. In *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, pages 2219–2225. IEEE, 2006.

[10] Manoj Srinivasan and Andy Ruina. Computer optimization of a minimal biped model discovers walking and running. *Nature*, 439:72–75, 2006.

[11] Zhenglong Sun and Nico Roos. An energy efficient gait for a nao robot. *BNAIC*, 2013.

[12] Zhenglong Sun and Nico Roos. An energy efficient dynamic gait for a nao robot. In *IEEE Conference on Autonomous Robot Systems and Competitions*, 2014.

Valuation of Cooperation and Defection in Small-World Networks: A Behavioral Robotic Approach

Bijan Ranjbar-Sahraei[a], Irme M. Groothuis[a], Karl Tuyls[b], Gerhard Weiss[a]

[a] *Department of Knowledge Engineering, Maastricht University*
[b] *University of Liverpool, Liverpool, United Kingdom*

Abstract

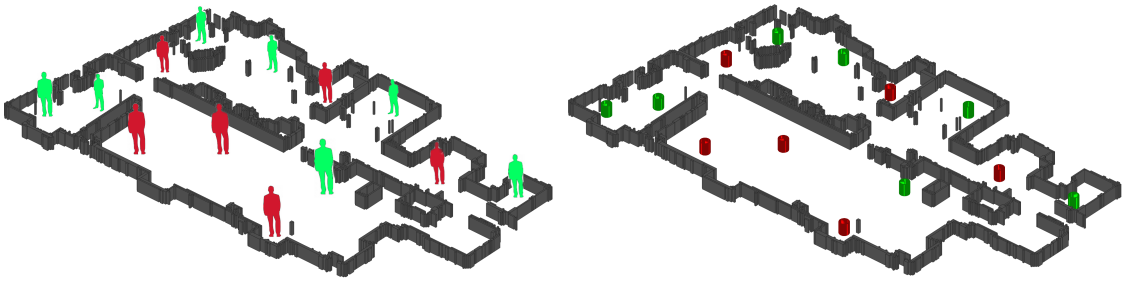
Valuation of behaviors in a social network is a very complex task due to dynamic nature of interactions, changes in behaviors and difficulties in defining the norms to evaluate the behaviors. Though, this valuation is a mandatory first step in studying evolution of behaviors in social networks. Therefore, in this paper two major game theoretical behaviors in social networks, namely *Cooperation* and *Defection* are programmed in members of a large group of robots. Various experiments on the multi-robot system are carried out to study the fitness of individuals who have adopted each of these behaviors in a *Small-World*-like environment compared to a *Regular* environment. The results of this study reveal one of the important characteristics of small-world networks in which individuals are not directly connected to one another, but have indirect links to every other via a small number of intermediate individuals: The more individuals adopt Cooperation in a small-world environment the less benefits they'll get in the group. In contrast, our results show that in regular environments where no short connection exists between most of the individuals, a reverse phenomenon is seen: The cooperators surpass defectors once they are in the majority. Such kind of results suggest that, by getting advantage of the proposed multi-robot framework, valuable contributions can be delivered to the field of social science.

1 Introduction

"Evolution of Cooperation" is a topic that seeks to provide an answer for why and how cooperators prevail in a society of selfish individuals. The systematic study of this topic dates back to the work of Axelrod and Hamilton in 1981 [3]. Despite of the very thorough study of this topic by Axelrod and Hamilton, many researchers from natural, social and computer sciences are still exploring this problem from different perspectives [14]. This shows the complexities and dimensions of this topic which is further increased by the emergence of social network models such as *Scale Free* and *Small-World* Networks [4, 19].

A central question in this line of research is which criteria play key roles in emergence and persistence of cooperation in social networks. For example in [17] the role of social diversities, such as diversity in number of interactions, in increasing the cooperative behaviors is studied. In [12], the role of commitment strategy in reaching certain levels of cooperating behaviors in the network is studied. More recently, in [15] a theoretical framework based on Control theory is proposed for evolution of cooperation, and in [16] the intertwined evolution of behaviors and emergence of networks are studied simultaneously.

The physical environment and external disturbances can also play a key role in evolution of behaviors in social networks. However, studying the role of such criteria using theoretical frameworks and numerical simulations is very difficult and even infeasible in many cases. In contrast, behavioral sciences have the capability to systematically observe the effects of physical environment on individual behaviors.. In



(a) Behavioral studies using human subjects (b) Behavioral studies using robotic agents

Figure 1: A group of agents interacting in an office-like environment. The color represents specific type of behavior, and as agents meet in the environment they interact based on their behaviors. Consequently, according to the payoff matrix of a Game Theoretical setting each individual receives a payoff in form of acceleration or punishment in form of delay in movement.

behavioral robotics, as an alternative approach to traditional behavioral sciences, robots can be used in experiments where the physical environment or properties of the physical environment are likely to influence the outcome of social behaviors [5, 13]. For instance, Keller et al. [13] have shown that robots can be used to study the self-organization, communication, the evolution of cooperative and competitive behaviors.

Therefore, in this paper we use a group of robotic agents to valuate the cooperation and defection behavior (i.e., the first step to study evolution of cooperation) in a small-world setting. A framework is proposed, as shown in Figure 1, in which robots move randomly in an environment. Whenever two robots are close to each other and are in line of sight of each other, they start playing a simple game in which each agent can either defect or cooperate. Based on the Prisoner's Dilemma game, the punishment and payoffs are embedded in form of delays and accelerations in movements, respectively. Consequently, the average speed of individuals is used to measure the fitness (value) of each individual in the group. Compared to existing multi-robot platforms for studying behaviors in social settings (e.g., in [5]), the proposed framework is easier to implement and is easier to be extended to other social experiments.

The rest of the paper is organized as following: First, the preliminary information are provided in Section 2, then the methodology used to propose the behavioral robotic approach is introduced in Section 3. Experiments on two different type of environments are provided in Section 4 and discussions on results and concluding remarks are provided in Sections 5 and 6.

2 Preliminaries

In this section we introduce the elementary background on Game Theory and Social Networks which have formed the foundations of this work.

2.1 Game Theory

Game theory models strategic interactions in the form of games. Each player has a set of actions, and a preference over the joint action space that is captured in the received payoffs. The goal for each player is to come up with a strategy that maximizes his expected payoff in the game. A strategy that maximizes the payoff given fixed strategies for all opponents is called a best response to those strategies. The players are thought of as rational, in the sense that each player purely tries to maximize his own payoff, and assumes the others are doing likewise. Under this assumption, the Nash equilibrium concept can be used to study what players will reasonably choose to do. A set of strategies forms a Nash equilibrium if no single player can do better by unilaterally switching to a different strategy [9].

The prisoner's dilemma is a canonical example of a game. In this game, two players have the option of either cooperating with the other or defecting. The players have to decide simultaneously what to do, given that they do not know their opponents decision. In this game, the best solution for both players is to cooperate and receive a low reward, however individually both are tempted by the higher payoff of defection, leaving the other with the sucker punishment. As both reason like this, they end up in the less favorable state of mutual defection, receiving a punishment, hence the dilemma.

2.2 Social Networks

Networks, in the most general sense, can be seen as patterns of interconnections between sets of entities [8]. These entities make up the nodes in the network, whereas the edges represent how those entities interact or how they are related. A social network is a type of network where the nodes represent a set of social agents (e.g., individuals in a society or firms in a market) and the interconnections between the nodes, represent a set of ties between these agents.

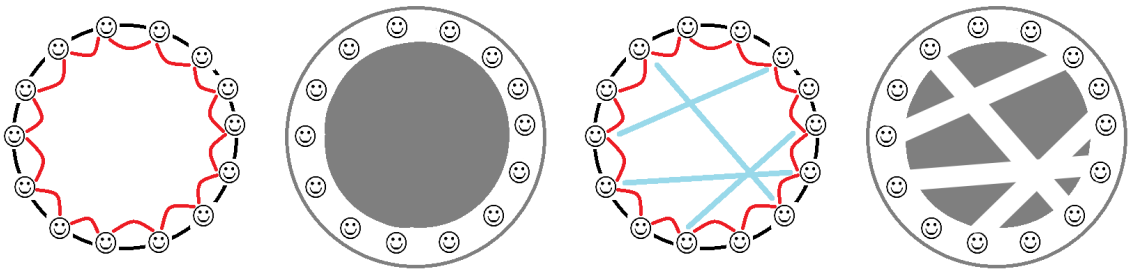
Based on the structural properties of a social network, different network models are already proposed: In a *regular* network all nodes have exactly the same degree (i.e., the same number of connections to other nodes). For instance, a ring is a regular network of degree 2, and a complete network is a regular network of degree $n-1$, where n is number of nodes. Another model used to describe real-world networks is the *small-world* model, that is characterized by short average path lengths between nodes and high clustering [19]. In small-world networks most nodes are not direct neighbors of one another but still can mostly be reached from every other by a small number of steps. A small-world network model provides a realistic representation of social networks.

3 Methodology

This section details the multi-robot behavioral framework proposed in this paper. First, the environment design inspired by social network topology is described, then the game theoretical definitions are provided, and finally the robot behavior design is explained.

3.1 Environment Structure

As described in Subsection 2.2, network models can represent the topological structure of relations between individuals. However, in real world environments where individuals continuously change their interactions, and consequently their neighborhood changes, such models cannot be directly used. Therefore, we propose an environment map corresponding to each network topology as shown in Figure 2.



(a) Regular Network (b) Regular Environment (c) Small World Network (d) Small World Environment

Figure 2: Network topologies and their corresponding environment map.

As can be seen in Subfigure 2(b), in a regular environment agents are more constrained to move within their local neighborhood; moving to far locations and meeting some other robots requires traversing a long

distance. In contrast, in Subfigure 2(d) the shortcuts between different regions of the small-world environment allow agents to reach different locations and meet most of other robots by traversing a short distance.

The proposed structures for environment map, namely *regular environment* and *small-world environment*, capture the main characteristics of their corresponding network model: A low number of immediate neighbors for individuals can be seen in both networks. Besides, most individuals need to traverse a long distance to meet each other in the former environment. In contrast, individuals can mostly meet each other by traversing the shortcuts provided in the latter environment.

3.2 Game Theoretical Definitions

In the proposed framework, each robot is considered as a *Player* and the whole experiment, in which robots move randomly and interact whenever they face each other, is considered as a multi-player *Game*. The *Strategy* of the players is defined as the main interaction policy that they use when they face each other:

- *Cooperation*: A cooperating robot avoids collision with other robots by turning and changing its movement direction.
- *Defection*: A defecting robot moves straight ahead toward other robots, and ignores the possibility of collision.
- *Punishment and Reward*: When robots collide with each other and get stuck, the amount of time it takes for them to continue moving is considered as a punishment. In contrast, when they avoid collision, the time they can save by moving straight ahead is considered as a reward.

The *fitness* of each robot is defined as the average absolute speed of the robot over time. Specifically, the fitness for the i^{th} robot at k^{th} time steps is given by

$$F_i = \frac{1}{k} \sum_{j=1}^{k} \frac{\left| P_i(j+1) - P_i(j) \right|}{\Delta t} \tag{1}$$

where $P_i(j)$ is the position of robot i at iteration j, Δt is, sufficiently small, equal to the time step in experiments, and $\left|.\right|$ denotes the distance between two points in a 2D plane.

3.3 Behavior Design

In order to implement the game theoretical setting introduced in previous subsection on robotic agents, robots should be programmed to follow a set of behaviors that each will be activated in different situations.

First of all every robot is programmed to do a *random walk* in the environment, while avoiding collisions with obstacles.

The second behavior is an *interaction behavior* which is activated whenever two robots face each other directly and are close enough to play the game. In this case two robots head for a single passway from opposite directions. Therefore, the first robot which swerves leaves the passway to the other robot. If neither robot swerves, robots will bump into each other (or at least get stuck in a very close distance). It is thus the best chance for defectors to stay straight while the cooperator swerves. Additionally, a crash is the worst outcome for two defectors. One can imagine a large group of cooperators can easily work in the environment, while a group of defectors will result in all robots clumped together, or different groups of robots clumped together in different areas of the environment.

The outcome of each 2-player game can be implicitly observed from the movements of individuals. A defector heading toward a cooperator can continue straight a way, while cooperators need to swerve, and two defectors bumping into each other will get stuck for a while, until they can both completely turn around and continue in different directions. This shows how the fitness definition in Eq. 1 can reflect the reward/punishment of each game.

The third behavior which is implemented on robots is an *escape* behavior which is activated when robots get stuck due to unknown reasons for a long time. This can happen due to errors in sensor reading, crowded passways, etc. In such situations, robots start to turn randomly and move back and forth, until they can continue with the *random walk* behavior. For more information about implementation of the behaviors see [10, 11].

4 Experiments

In this section, groups of cooperating and defecting robots are used to evaluate each of these behaviors in two different types of regular and small-world environments.

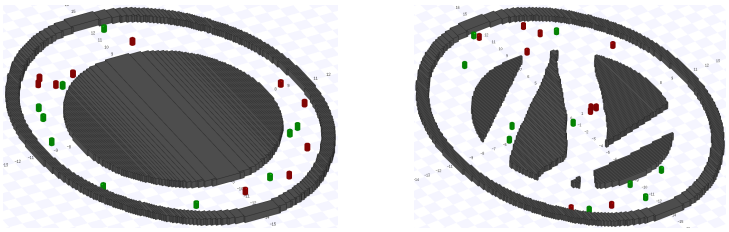
4.1 Robotic Platform and Simulation Environment

The robots used for the proposed robotic platform should be able to do basic tasks such as moving in different directions with a reasonable speed and detection of obstacles. Robots should also be able to interact with other near by robots following their pre-defined behaviors (i.e., either cooperation or defection), and avoid getting stuck in the environment due to unknown reasons.

Therefore, this work uses the *Turtlebot*[1] robot. This robot is equipped with strong wheels and encoders, a Kinect[2] camera and a laptop. This robot can be easily programmed by ROS[3], which allows us to implement a specific code easily on multiple robots. Besides, the exact same code can be used in simulation environment. For more sophisticated details about Turtlebot features and its comparison with a limited-resource robot, refer to [1].

For the experiments reported in this paper, identical Turtlebot robots are defined in the Stage simulator [18]. This simulator is designed for simulation of multi-agent autonomous systems, and it allows us to modify the environment maps easily using a Portable Gray Map (PGM) format file.

Every experiment is carried out with 20 robots with pre-defined behaviors, starting from a random initial position in the environments shown in Figure 3. Each experiment is executed two times for a duration of 5 minutes each.



(a) Regular Environment (b) Small World Environment

Figure 3: The environment map used for experiments. (a) a regular map in which robots are surrounded by their immediate neighbors and can hardly move to other neighborhoods. (b) a small-world map in which robots have shortcuts for moving to many different locations.

In all experiments, the diameter of the environment is $25m$. The laser scanner of the robots detects the range $0 - 4m$, its field of view is 170 degrees with 340 rays. The robot has the dimensions $35cm \times 35cm \times 45cm$, which is identical to a real Turtlebot.

[1] http://www.turtlebot.com
[2] http://en.wikipedia.org/wiki/Kinect
[3] Robot Operating System http://www.ros.org

4.2 Results

Seven main experiments are defined for evaluating the Cooperation and Defection in a social structure (Table 1). These experiments are different in terms of ratio between cooperators and defectors in the multi-robot group (ranging from 5% to 95%).

Table 1: Valuation of Cooperation and Defection in different experiments. F_c and F_d denote the average fitness of cooperators and defector, respectively.

Experiment	Number of Cooperators	Number of Defectors	Regular Environment			Small-World Environment		
			F_c	F_d	F_c/F_d	F_c	F_d	F_c/F_d
Exp. 1	1	19	0.086	0.142	0.6056	0.419	0.119	3.5210
Exp. 2	5	15	0.173	0.126	1.3730	0.310	0.166	1.8675
Exp. 3	8	12	0.160	0.118	1.3559	0.295	0.165	1.7879
Exp. 4	10	10	0.182	0.090	2.0222	0.299	0.172	1.7384
Exp. 5	12	8	0.180	0.064	2.8125	0.278	0.241	1.1535
Exp. 6	15	5	0.252	0.082	3.0732	0.334	0.411	0.8127
Exp. 7	19	1	0.255	0.099	2.5758	0.370	0.423	0.8747

The results of the experiments are illustrated in Subfigures 4(a) and 4(b). Subfigure 4(a) shows how the average fitness of defectors and cooperators change with respect to the ratio of (number of) cooperators to (number of) defectors. As can be seen in this subfigure, by increasing the ratio of cooperators to defectors, the fitness of defectors F_d decreases compared to fitness of cooperators F_c in a regular environment. In contrast, in the small-world network the reverse behavior is observed, and defectors can even get a higher fitness if they are in minority compared to cooperators.

Given that the average fitness represents the average speed of the robots, in a regular environment, robots can move with an average speed of $15cm/s$ and this increases to about $25cm/s$ in a small-world environment.

The ratio of F_c to F_d is depicted in Subfigure 4(b). This subfigure, clearly shows the reverse trend between regular environment and small-world environment.

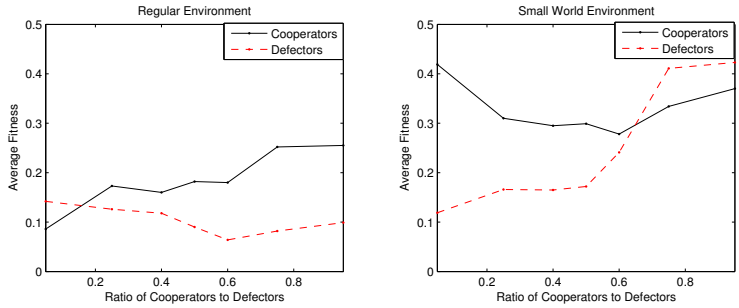
Based on the presented results in this subsection, in a regular environment the value of cooperation compared to defection is increased by an increase in number of cooperators. In contrast, in a small-world environment, the value of cooperation compared to defection is decreased by an increase in number of cooperators. To sum up, in small-world networks the individuals whose behavior is in minority get the highest payoff, while in regular networks the individuals in majority can get a higher payoff.

Another interesting result from Figure 4 is that the macroscopic fitness of the group (i.e. the sum of fitness of all individuals) is proportional to number of cooperators; the more cooperators exist in the group the faster robots can traverse the environment, and higher payoff they get.

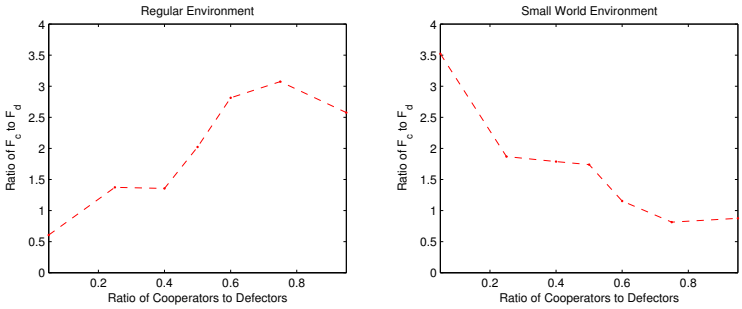
5 Discussion

The results provided in previous section clearly show that the proposed multi-robot framework is capable of studying the simple behaviors in a social network. For instance, a major difference between observations on a regular network and small-world network that was already reported elsewhere [19] is confirmed through our experiments. Furthermore, the effect of population ratio on the fitness levels which is a known concept for the Predator-Prey scenarios [2] is also confirmed by this work.

One example of unexpected behaviors observed during the experiments, is the chaotic behavior in dense clusters of robots. In such situations, robots can hardly determine a fixed opponent robot to play the game with, and many unexpected crashes take place. Our analysis revealed that cooperators behave like defectors in such chaotic situations, and therefore decreases the fitness of every individual in that cluster. Such dense clusters are seen more often in regular environments than small-world ones, as in the latter one there are



(a) Average fitness of cooperators and defectors



(b) Ratio of fitness of cooperators to defectors

Figure 4: A comparison between fitness of cooperators and defectors in different experiments.

usually shortcuts for robots to escape from such dense clusters. This can be considered as a main reason for the reverse trends between regular and small-world environments as shown Figure 4.

The discussed research opens several interesting research avenues. For instance, assume that following the "survival of the fittest rule" [7], we let the robots learn from the behavior of the fittest in the group. As explained in this paper the defectors in minority can be the fittest in a small-world environment. As other individuals start to learn this behavior, their fitness will decrease. Given that the group should eventually agree on a fixed strategy [15], studying which strategy the group agrees on can be very interesting. Furthermore the feasible ways to influence the behaviors [6] can be studied using the proposed framework. These two clearly explain the next steps in our study.

6 Conclusion

Due to complexities of the social ties in a network of social actors, valuation of social behaviors is a very difficult task. The effects of physical environment and interaction dynamics on such behaviors can make it almost impossible to use theoretical approaches and numerical simulations to analyze evolution of behaviors. In this paper a behavioral robotic approach was proposed which maintains the formal game theoretical definitions, and is still feasible for implementation on a large group of robots. The experimental results on two different environment structures with various population distributions for cooperating and defecting agents verified that the proposed framework generates a realistic valuation for behaviors and can be used for further studies on social behaviors. For instance, the results showed that defecting agents which are in the minority compared to cooperating agents can achieve a high fitness in a small-world setting, but this is not the case in regular networks due to emerged chaotic behaviors.

References

[1] Sjriek Alers, Karl Tuyls, Bijan Ranjbar-Sahraei, Daniel Cleas, and Gerhard Weiss. Insect-inspired robot coordination: Foraging and coverage. In *14th International Conference on the Synthesis and Simulation of Living Systems (ALIFE 14)*, 2014.

[2] Roger Arditi and Lev R Ginzburg. Coupling in predator-prey dynamics: ratio-dependence. *Journal of Theoretical Biology*, 139(3):311–326, 1989.

[3] Robert Axelrod and William D Hamilton. The evolution of cooperation. *Science*, 211(4489):1390–1396, 1981.

[4] Albert-László Barabási et al. Scale-free networks: a decade and beyond. *science*, 325(5939), 2009.

[5] Andreas Birk and Julie Wiernik. An N-player prisoner's dilemma in a robotic ecosystem. *Robotics and Autonomous Systems*, 39(3):223–233, 2002.

[6] Daan Bloembergen, Bijan Ranjbar-Sahraei, Haitham Bou Ammar, Karl Tuyls, and Gerhard Weiss. Influencing social networks: An optimal control study. *Proceedings of the 21st European Conference on Artificial Intelligence (ECAI 2014)*, 2014.

[7] Eddie Dekel and Suzanne Scotchmer. On the evolution of optimizing behavior. *Journal of Economic Theory*, 57(2):392–406, 1992.

[8] David Easley and Jon Kleinberg. *Networks, Crowds, and Markets: Reasoning about a Highly Connected World*. Cambridge University Press, 2010.

[9] R. Gibbons. *A Primer in Game Theory*. Pearson Education, 1992.

[10] Irme M. Groothuis. *Studying Social Interactions using Swarm Robotics*. Department of Knowledge Engineering, Maastricht University, 2014. Bachelor's Thesis.

[11] Irme M. Groothuis, Bijan Ranjbar-Sahraei, Karl Tuyls, and Gerhard Weiss. Demonstration video for studying social interactions using swarm robotics. `http://swarmlab.unimaas.nl/papers/bnaic-2014-demo/`. Accessed 2014-07-07.

[12] The Anh Han, Luís Moniz Pereira, Francisco C Santos, Tom Lenaerts, et al. Good agreements make good friends. *Scientific reports*, 3, 2013.

[13] Laurent Keller, Steffen Wischmann, Dario Floreano, Sara Mitri, et al. Using robots to understand social behavior. Technical report, Wiley-Blackwell, 2012.

[14] Martin A Nowak. Five rules for the evolution of cooperation. *science*, 314(5805):1560–1563, 2006.

[15] Bijan Ranjbar-Sahraei, Haitham Bou Ammar, Daan Bloembergen, Karl Tuyls, and Gerhard Weiss. Theory of cooperation in complex social networks. In *Proceedings of the 25th AAAI Conference on Artificial Intelligence (AAAI-14)*, 2014.

[16] Bijan Ranjbar-Sahraei, Daan Bloembergen, Haitham Bou Ammar, Karl Tuyls, and Gerhard Weiss. Effects of evolution on the emergence of scale free networks. In *The 14th International Conference on the Synthesis and Simulation of Living Systems (ALIFE 14).*, 2014.

[17] Francisco C Santos, Flavio L Pinheiro, Tom Lenaerts, and Jorge M Pacheco. The role of diversity in the evolution of cooperation. *Journal of theoretical biology*, 299:88–96, 2012.

[18] Richard T Vaughan. Stage: A multiple robot simulator. Technical report, Technical Report IRIS-00-394, Institute for Robotics and Intelligent Systems, University of Southern California, 2000.

[19] Duncan J Watts and Steven H Strogatz. Collective dynamics of small-world networks. *nature*, 393(6684):440–442, 1998.

Validating Ontologies for Question Generation

Marten Teitsma [a] Jacobijn Sandberg [b] Bob Wielinga [c]
Guus Schreiber [c]

[a] *Amsterdam University of Applied Sciences, The Netherlands*
[b] *University of Amsterdam, The Netherlands* [c] *VU University Amsterdam, The Netherlands*

Abstract

In this paper we present an experiment which has been performed to validate a pragmatic-based, expert-based and basic-level ontology. These ontologies were created for use in an application which generates questions for ordinary people with the purpose to determine a crisis situation. All three ontologies have specific characteristics related to their method of creation. This experiment shows that using the basic-level ontology results in the fastest and least ambiguous determination of a crisis situation.

1 Introduction

Making use of humans to gather information is the central subject in the new emerging field of Human-Centered Sensing (HCS) [2]. The application we propose here is typified as a participatory sensor because humans are producing information and not just facilitating the gathering of data as in opportunistic sensing e.g., a mobile device recording background noise. By answering questions the human observers can help making clear what the situation is. Research on crisis situations shows a variety of public involvement during a crisis. Not only experts in crisis management convey information about the crisis at hand but also ordinary people, i.e. people with no specific knowledge of the situation they describe. It becomes more and more accepted to regard members of the public as true 'first responders' [4].

To determine a crisis situation we use the Situation Awareness Question Generator (SAQG) which automatically generates questions from an ontology. During this experiment questions were generated by asking after the specification of a concept, i.e. which of the subordinate concepts is a more specific description of an object in the real world. These questions are presented to ordinary people who, by answering the questions, help to determine the situation. In this paper we show an experiment with participants who determine what kind of object is on fire.

The SAQG is installed on a mobile device and generates questions from an ontology which is received from a server. These questions are presented to the user of the application. The answer given by the user is computed by SAQG and gives rise to another question until a situation is determined by the application. This situation is then communicated to the server. Previous experiments showed that questions generated from an ontology created by knowledge engineers (an expert-based ontology) did not lead to thrustworthy answers while questions generated from a ontology based on pragmatic considerations (a pragmatic-based ontology) were suitable to determine a situation [6, 5].

We discern three sources for the creation of ontologies: *a*) pragmatic classifications found in a particular domain, *b*) existing expert-based ontologies and *c*) natural categorization with basic-level concepts. We developed methods to create these ontologies. To find the ontology most suitable for generating questions we measured the three ontologies using four criteria: *a*) the ontology must have a structure which is useful for the task at hand, i.e. question answering on a mobile device, *b*) the construction of the ontology is efficient, *c*) the ontology must be complete, i.e. all concepts that are relevant should be contained in the ontology and *d*) the ontology should be compliant with human thought.

We used several metrics to compare these ontologies which showed that expert-based ontologies are most easy to construct but lack required cognitive ergonomic characteristics. Basic-level ontologies have structure and concepts which are better in terms of cognitive ergonomics but are most expensive to construct [7]. In this paper we present an experiment in which a simulation of a crisis situations is created and where participants help to determine that situation. With this experiment we identify the most suitable ontology for such a task.

Characteristics of the ontologies are presented in Section 2. The experiment we conducted and the results are presented in Section 3. The results are discussed in Section 4 and conclusions are drawn in Section 5.

2 The ontologies

The ontologies used by SAQG are composed of a representational part, a generic part and a domain specific part [8]. The representational and generic part of the ontology are a revised version of the Situation Theory Ontology [3]. The domain specific part captures the relevant knowledge of a particular domain. For all three ontologies we have defined a backbone consisting of the concepts *Streetobject*, *Ship*, *Roadvehicle*, *Railvehicle*, *Nature*, *Industry*, *Building*, *Aircraft* which represent the subjects we are interested in.

The ontologies we use in the experiments were created with three different methods [7]. The pragmatic based ontology (pbo) was developed from a classification used by the fire department at an emergency call center. Relations are heterogeneous and terms are functionally not the same in this classification. To make it suitable for automatic reasoning some knowledge engineering had to be done. The expert-based ontology (ebo) was constructed from existing ontologies created by domain experts. We extracted concepts related to the backbone from AATNed [1] and Cornetto [9] and merged these concepts into an ontology. Because the result was a very large ontology we filtered out less frequently used concepts. The basic-level ontology (blo) is an ontology based on empirical data elicited from ordinary people. In several experiments we asked for concepts related to the backbone and visual properties of objects denoted by these concepts. We then used the properties of the concepts to automatically create a hierarchy of concepts. This algorithm made use of the psychological phenomenon of basic-level concepts. The process to create the blo was rather laborious.

To make the ebo more efficient and the blo suitable for use with SAQG these ontologies were re-engineered. The ebo was significantly improved by applying some small changes. The most important of these improvements was the reduction of path length for concepts which only had one subordinate concept. The superordinate concept was then replaced by the subordinate concept. The way we use an ontology in our application, i.e. asking a question and suggesting answers, does not generate information when only one answer is possible. We joined some synonyms and some concepts were replaced as subordinate to another concept when this seemed appropriate. The blo was not suitable due to the method to create this ontology: most superordinate concepts did not have a meaningful label. To create labels for the superordinate concepts we made use of guidelines formulated by van Heijst [8]. It is clear that the most perfect label consists of one word. Unfortunately it was hard to find appropriate words for some of the superordinate concepts and we had to be content with compound labels for some concepts. Only the pragmatic-based ontology could be used without modification by the application. The ontologies created were given new names by adding the prefix 'new': new expert-based ontology (nebo) and new basic-level ontology (nblo).

In the experiment as described in the next section we present participants with situations in three domains: *Buildings*, *Road vehicles* and *Water vehicles*. In Table 1 some characteristics of these subtrees in each ontology are shown. The subtrees of the nebo are largest and also have the highest average path length. The average number of subclasses for the pbo is highest. With respect to entropy the subtrees of the nebo have the highest value compared with the subtrees of the pbo and nblo. The subtrees for *Building* of the nebo and the nblo show a remarkably higher entropy than for other subtrees. The correlation between entropy for the subtrees of all three ontologies and the number of concepts is strong: $r(7) = .97, p < .001$.

		pbo	nebo	nblo
Buildings	number of concepts	15	198	56
	average path length	1.71	3.68	2.72
	average number of subclasses	5.33	3.26	3.56
	entropy	2.67	9.53	5.74
Water vehicles	number of concepts	20	127	21
	average path length	1.77	4.08	3.04
	average number of subclasses	7.00	4.40	2.75
	entropy	2.94	7.50	3.92
Road vehicles	number of concepts	15	97	22
	average path length	1.71	3.59	2.88
	average number of subclasses	5.33	4.59	3.83
	entropy	2.67	7.02	3.80

Table 1: Metrics of the subtrees

3 Experiment

3.1 Method

The experiment was done in eight sessions with a total of 110 participants. The smallest session was done with 5 participants, the largest with 23 participants. All participants were students of the Amsterdam University of Applied Sciences and between 18 and 22 years of age.

We used three videos which showed an object on fire in the domains *Building* (video 1), *Water vehicle* (video 2) and *Road vehicle* (video 3). All the videos were cut to one minute and stripped of sound. Each participant only used one ontology while observing the successively presented situations to prevent interference between the use of different ontologies. The videos were shown to all the participants and each was assigned one of the three ontologies.

For instruction we used an instruction sheet which presented the participants the goal of the experiment, how SAQG uses an ontology to generate questions and the sequence of steps to make the application work, e.g. making connection to the Internet. During the instruction particular attention was given to the possibility of backward navigation, i.e. return to a previous question and the possibility of choosing a superordinate concept when the subordinate concepts are not known or suitable. All the participants were using the same mobile device: an IDEOS X5 with Android 2.2.1. Because the mobile device we used was the same for all participants and most probably different from their own, the participants were given a small amount of time to get used to the mobile device.

We started with a video about an airplane on fire. The results of the determination done by the participants were not used for the experiment. This was done to get an equal starting point for the participants for each video. Otherwise the participants would have to get used to the question answering after seeing video 1 which they would not have to after seeing video 2 and 3. Because all the ontologies did have the same backbone the first question was 'What is on fire' with the same multiple choices for all the ontologies (the top of the subtrees for all the ontologies was the same (see Figure 1).

We logged the data to gather results and create statistics. To gather additional data we developed a questionnaire. After each video we let the participants answer some questions about the application. We asked the following questions: 'Do you miss a concept which describes the object on fire better?', 'What do you think of the sequence of questions?', 'Do you understand all the used concepts?' and 'What do you think of the Graphical User Interface?'. The first question could be answered by yes or no. The other questions could be answered by choosing from a scale of 1 to 5 a number which reflected their evaluation on this topic, where 1 was a negative evaluation and 5 a positive evaluation.

To measure whether a participant chose the right concept for the object on fire which was shown on the video, we developed three metrics. For the first measure we created a 'gold standard'. We observed, as knowledge experts, independent from each other, the presented situations and chose concepts we thought best represented the object on fire shown on each video independent of any ontology. Then we discussed our own preferences and agreed on one concept for each video as being the best representation of the object on fire.

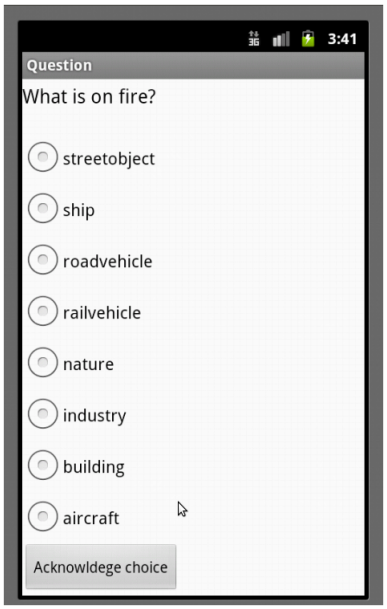


Figure 1: First screen generated from the ontology

For the second measure a concept from each ontology was drawn. This concept was similar to the preferred concept or had a meaning as closely as possible to this concept. We also chose some concepts closely representing the object on fire, i.e. alternatives. In a discussion we determined the perfect answer and the alternative concepts. To compare the results on this measurement per ontology we gave each answer a value: the best answer was given one point and the alternatives half a point.

For the third measure we measured how many different concepts were chosen by the participants, i.e. the variability of the chosen concepts, and how often the most chosen concept was chosen relative to all other choices, i.e. the relative frequency of the most often chosen concept. We expect consensus among the participants about the object on fire for each ontology used after watching the three videos. The level of agreement among the participants indicates the suitability of the ontology for this task.

3.2 Results and analysis

How long it took for the participants to determine the object on fire is shown in Table 2. The participants which used the nebo on average needed much more time than the participants which used the pbo or the nblo. The difference between the time needed to determine the object on fire when using the pbo or the nblo was not significant (two-sample $t(210) = .6, p = .53$). The correlation between the time needed to determine the object on fire and the entropy of the subtree as shown in Table 1 is significant: $r = .73, n = 9, p < .05$. The correlation with the number of concepts is even a bit stronger: $r = .77, n = 9, p < .05$.

	pbo	nebo	nblo
n	36	35	39
video 1	29.54(20.97)	64.73(26.59)	40.49(16.72)
video 2	43.55(34.75)	80.59(39.53)	37.97(31.65)
video 3	25.97(16.35)	34.25(21.15)	26.77(10.48)
mean	33.02(26.16)	59.86(35.52)	35.08(22.17)

Table 2: Mean duration of determination and standard deviation for each video and ontology

Table 3 shows how the ontologies relate to each other when the mean duration of determination for each video is corrected by the average path length of the subtree which was used. The correlation of the mean duration of determination for each video corrected by the average path length with the time needed

to determine an object on fire is not significant. The correlation between the path length as shown in Table 1 and the time needed to determine an object is significant: $r = .68, n = 9, p < .05$.

	pbo	nebo	nblo
video 1	17.28(21.05)	17.47(18.78)	14.89(15.62)
video 2	24.61(30.72)	19.72(21.79)	12.49(16.12)
video 3	15.19(17.43)	9.60(11.21)	9.30(9.88)
mean	19.02(23.07)	15.59(17.26)	12.22(10.54)

Table 3: Mean duration of determination for each video and ontology corrected by the average path length of the subtree

The time needed to determine the object on fire corrected by the number of subclasses is shown in Table 4. The correlation with the time needed to determine an object on fire is rather high: $r = .86, n = 9, p < .05$. The correlation of the number of subclasses of a subtree and the time needed to determine an object using that subtree is not significant: $r = .17, n = 9, p = .66$.

	pbo	nebo	nblo
video 1	5.54(6.75)	19.72(21.20)	11.37(11.94)
video 2	6.22(7.77)	18.28(20.21)	13.81(17.82)
video 3	4.87(5.59)	7.51(8.77)	6.99(7.43)
mean	5.56(6.70)	15.17(16.73)	10.72(12.40)

Table 4: Mean duration of determination for each video and ontology corrected by the number of subclasses of the subtree

In Table 5 the results from the questionnaire are shown. The participants did not miss concepts in the determination process in one ontology more than in another. Taken all ontologies together nearly 63% of the participants did not miss a concept to describe the object on fire. The pbo, which is the smallest ontology, scores worst on this question.

		1	2	3	4
pbo	average	0.59	4.22	4.15	2.83
n = 36	video 1	0.611	4.36	4.36	2.83
	video 2	0.81	3.94	3.44	2.83
	video 3	0.36	4.36	4.64	2.83
nebo	average	0.67	4.03	3.51	3.26
n = 35	video 1	0.69	4.20	3.40	3.29
	video 2	0.49	3.57	3.03	3.17
	video 3	0.83	4.31	4.11	3.31
nblo	average	0.68	4.30	4.44	3.29
n = 39	video 1	0.72	4.23	4.36	3.28
	video 2	0.90	4.36	4.26	3.28
	video 3	0.41	4.31	4.69	3.31

Table 5: Results from questionnaire. 1: 'Do you miss a concept which describes the object on fire better?' (no(1)/yes(0)), 2: 'What do you think of the sequence of questions?' (1-5), 3: 'Do you understand all the used concepts?' (1-5), 4: 'What do you think of the Graphical User Interface?' (1-5).

Comparing the mutual results of the questionnaire we see a difference in mean for the comprehension of concepts between the nebo (3.51) and nblo (4.44) which is significant ($F = 5.77, p < .05$). The difference in mean for the GUI between the pbo (2.83) and nebo (3.26) is significant ($F = 4.71, p < .05$).

When we compare the results as shown in Table 2 with the results on the question whether the participant understands the concepts a significant correlation occurs: $r = .91, n = 9, p < .001$. When

comparing the time needed to determine an object with the results on the evaluation of the sequence of questions also a significant correlation can be found: $r = .83, n = 9, p < .05$. The correlation between the two questions just mentioned is also significant: $r = .84, n = 9, p < .05$. A comparison between the values for the question after the Graphical User Interface and the number of subclasses for each subtree shows a significant correlation: $r = .84, n = 9, p < .05$.

	independent of ontologies			dependent on ontologies		
	pbo	nebo	nblo	pbo	nebo	nblo
video 1	0	6	0.90	31.11	10.50	16.71
video 2	13.61	0	24.23	21.88	6.50	27.37
video 3	0	29	0	35.00	29.00	33.21
total	13.61	35	25.13	87.99	46.0	77.29

Table 6: Validation of the ontologies using the gold standard (normalized) independent and dependent on ontologies

Table 6 shows the score for each ontology using the gold standard created independent of the ontologies. For each time a participant chose the preferred concept an ontology scores a point and when an alternative was chosen half a point was given. To compare the ontologies a normalization has been applied because the number of participants using the pbo was 36, the nebo 35 and the nblo 39, the result was corrected by respectively $\frac{35}{36}$, 1 and $\frac{35}{39}$. It is clear that nebo scores best and pbo worst using this measurement. For participants which used the pbo it was not possible to choose the right concept when seeing video 1 and 3. The same holds for the nebo with video 2 and the nblo with video 3. The participants which used the nblo with video 1 who wanted to choose *Bungalow*, had to choose *Holiday accommodation* (*Vakantieverblijf*) first instead of *Residence* (*woning*). Most participants chose *Residence*. Table 6 also shows the score for each ontology using the gold standard and alternatives created with use of concepts which are part of the ontologies, i.e. dependent on ontologies. The lowest score is for nebo. The best score is for pbo. Due to the strong variability of choices made by participants using the nebo, this ontology does not score high on video 1 and 2. Only with video 3 the nebo scores nearly as high as pbo and nblo. The variability of choices for all ontologes after seeing video 3 is much smaller.

Subject	pbo		nebo		nblo	
	#	%	#	%	#	%
Buildings	2	77.78	11	22.86	5	89.74
Water vehicles	6	38.89	12	37.14	5	69.23
Road vehicles	1	100.00	4	80.00	2	94.87
Average	3.00	72.22	9.00	46.67	4.00	84.61

Table 7: Variability (#) and relative frequency of the most often chosen concept (%)

Table 7 shows how many different choices were made by the participants (variability) and the mode relative to the total number of choices, i.e. how often the participants chose for the most frequently chosen concept, relative to the total number of choices made. The pbo scores best with respect to the variability. The best score is for nblo and the worst score for nebo with respect to the relative mode.

4 Discussion

We found five characteristics of the ontologies which have an influence on the the time needed to determine an object: path length, comprehension of concepts, sequence of questions, number of concepts and entropy. The path length of an ontology has a direct influence on the time needed to determine an object because it is a measure of the number of questions asked. Whether participants comprehend concepts has a less direct influence and is probably dependent on the familiarity of the participant with a particular domain, which can vary. The reason why the sequence of questions has influence is less clear. It might be a derivative of the number of questions and thus the path length. The number of concepts

and entropy have a strong correlation although the measurement of entropy also incorporates path length and number of subclasses.

The analysis of the choices made by the participants which was done using a gold standard shows rather confusing results. Using a gold standard set up by knowledge experts independent of the ontologies shows a best score for nebo, using a gold standard of concepts drawn from the ontologies shows a best score for pbo. To make matters even more confusing, when only the choices made by the participants are taken into account the best score is for nblo.

The gold standard independent of the ontologies is, to our opinion, a measurement for evaluating the completeness of an ontology, i.e. the number of concepts from an ontology also found in a corpus relative to the total number of concepts in that corpus, i.c. natural language. It makes clear the difference of determination of an object on fire by knowledge engineers on the one hand and ordinary people on the other. The gold standard drawn from the ontologies does measure the suitability of the ontologies for use by ordinary people much better. The outcome would even be similar to the outcome of Table 7, which shows the consensus among the participants, when for video 1 (*Building*) the gold standard would have been *Residence* (*Woning*) instead of *Bungalow*. The choice for *Bungalow* was again due to our own knowledge engineering. Erroneously omitting the concept *Cruiseship* from nebo did not have an effect on the overall results. The variability, as shown in Table 7, is the lowest for the pbo and highest for the nebo showing that the last ontology possibly offered too many concepts for the participants to be clear about which object was on fire. The pbo did offer much less concepts and scored better in this respect. The nblo scored nearly as good as the pbo on variability. The measurement of the relative frequence, as is shown in Table 7, shows for the nblo the best score although the pbo scored on video 3 an ideal score and was overall nearly as good as the nblo. With this measurement nebo scores worst, again. A measurement of the relative frequency is, in our opinion, most informative about the suitability of the ontologies for the task we envisage for it measures the unanimity among the participants about the object on fire better than the variability because this only shows how many different concepts are chosen and not how many times the most chosen concept was chosen.

5 Conclusion

In this paper an experiment is presented which was conducted to measure whether one of three ontologies is most suitable for the task of answering of automatically generated questions by ordinary people for determination of a situation. The three ontologies were each developed in a different way. Before we used the ontologies, two of the ontologies had to be re-engineered.

The participants understand the concepts used in the pragmatic-based ontology but evaluate the Graphical User Interface, relative to the nebo, negative. This is probably due to the large average number of subclasses of the pbo. The pbo scores best on the time needed to determine an object on fire (but the difference with the time needed when using the nblo is not significant). Using the 'gold standard' independent of the ontologies the pbo scores worst but when using the 'gold standard' with concepts drawn from the ontologies the pbo scores best. The pbo shows the smallest variability of choices when determining an object on fire. The average relative frequence of the most often chosen concept of the pbo is rather high but not the highest.

The concepts used in the new expert-based ontology are the least understood relative to the nblo. Using the nebo it took the participants the most time to determine an object on fire. Using the 'gold standard' independent of the ontologies the nebo scores best but when the concepts are drawn from the ontologies the nebo scores worst. The nebo shows the largest variability and the highest relative mode.

The participants do understand the concepts used in the new basic-level ontology rather well. When comparing the time needed to determine an object on fire the nblo scores nearly as good as the highest scoring ontology. The measurement using the 'gold standard' shows for nblo a mediocre result. The variability is rather low but not the lowest. The average relative mode is the highest of the three ontologies.

When predicting the time needed to determine an object on fire the average path length and the number of concepts used in an ontology are good indicators. Path length has a rather strong correlation with the time needed to determine an object. The number of subclasses does not have a correlation with the time needed to determine an object. Comprehension of the concepts also has a strong influence on the time needed to determine an object but is dependent on the specific knowledge of a domain users

have.

We conclude that each of the ontologies has its own merits and idiosyncrasies. When a very accurate determination of an object is required and users have expert knowledge about the domain, one should use the nebo. A longer time to determine such an object has to be accepted. The nblo scores nearly as good or better than the pbo on most respects, such as time needed to determine an object and consensus about the concept which denotes the object on fire best. The nblo is evaluated more positive than other ontologies with respect to the user interface. A great disadvantage is the high costs to develop a nblo. When an application such as SAQG is deployed on a large scale for many people such an effort could be worthwhile. The concepts and their categorization represented in the ontology used by SAQG to generate questions and possible answers should maximally comply with the language used by ordinary people. The new basic-level ontology has the best results in this respect.

References

[1] AATNed. *http://www.aat-ned.nl/*, 2012.

[2] M. Jiang and W.L. McGill. Participatory Risk Management: Managing Community Risk Through Games. In *Social Computing (SocialCom), IEEE Second International Conference on*, pages 25–32. IEEE, 2010.

[3] M.M. Kokar, C.J. Matheus, and K. Baclawski. Ontology-based situation awareness. *Information Fusion*, (10):83–98, 2009.

[4] L. Palen, K.M. Anderson, G. Mark, J. Martin, D. Sicker, M. Palmer, and D. Grunwald. A vision for technology-mediated support for public participation & assistance in mass emergencies & disasters. In *Proceedings of the 2010 ACM-BCS Visions of Computer Science Conference*, page 8. British Computer Society, 2010.

[5] M. Teitsma, J. Sandberg, M. Maris, and B. Wielinga. Using an Ontology to Automatically Generate Questions for the Determination of Situations. In *Database and Expert Systems Applications*, pages 456–463. Springer, 2011.

[6] M. Teitsma, J.A.C. Sandberg, M. Maris, and B.J. Wielinga. Automatic question generation to determine roles during a crisis. In *SOTICS 2011, The First International Conference on Social Eco-Informatics*, pages 37–42, 2011.

[7] Marten Teitsma, Jacobijn Sandberg, Guus Schreiber, Bob Wielinga, and Willem Robert van Hage. Engineering ontologies for question answering. *Applied Ontology*, 2014.

[8] G.A.C.M. van Heijst. The role of ontologies in knowledge engineering. 1995.

[9] P. Vossen, I. Maks, R. Segers, and H. van der Vliet. Integrating lexical units, synsets and ontology in the Cornetto Database. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*, 2008.

Monte-Carlo Tree Search for Poly-Y

Lesley Wevers Steven te Brinke

University of Twente, The Netherlands

Abstract

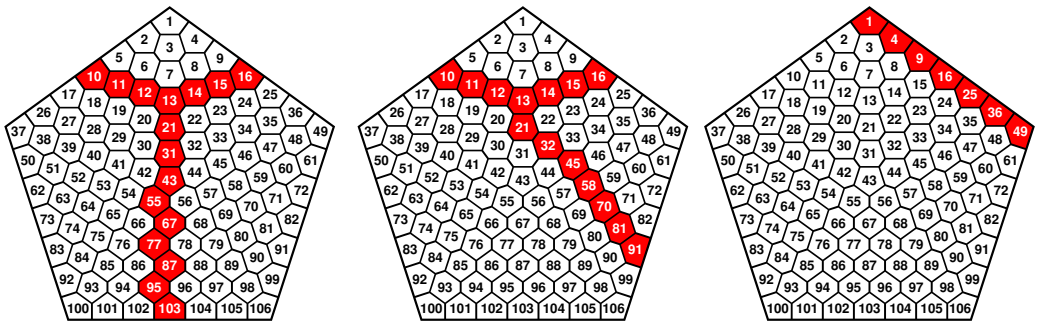
Monte-Carlo tree search (MCTS) is a heuristic search algorithm that has recently been very successful in the games of Go and Hex. In this paper, we describe an MCTS player for the game of Poly-Y, which is a connection game similar to Hex. Our player won the CodeCup 2014 AI programming competition. Our player uses MCTS with the all-moves-as-first heuristic, and detects basic heuristic patterns to defend virtual connections in Poly-Y. In the CodeCup, we can only use 30 s single-core computation time per game, whereas in Hex, 5 to 30 minutes of multi-core computation time is common. We improve the performance of our player in the early game with an opening book computed through self play. To assess the performance of our heuristics, we have performed a number of experiments.

1 Introduction

We created a program that plays Poly-Y, which is a generalization of the game Y, which in turn is a generalization of Hex. Our program won the CodeCup[1] 2014, which is an annual international AI board game programming competition organized by the Dutch Olympiad in Informatics. In contrast to other AI programming competitions, the CodeCup has major restrictions in computing resources: Programs only get 30 seconds of total playing time, 64 megabytes of memory, and cannot use multi-threading. This requires programs to use lightweight techniques for gameplay. The source code of our program (Lynx) is freely available[2], and it is possible to play against our program online[3].

Poly-Y In contrast to Y, which is played on a three-sided board, a Poly-Y board can have any odd number of sides greater or equal to three. An (n, r)-board consists of an n-sided center cell, with r rings of cells around it. For the CodeCup, we played on a $(5, 7)$-board, which has 106 cells (see Figure 1).

[1] CodeCup website: `http://www.codecup.nl/`
[2] Lynx source code: `https://github.com/lwevers/lynx/`
[3] Online playable version of our player: `https://maksverver.github.io/lynx/` (thanks to Maks Verver)



(a) Captures one corner **(b)** Captures two corners **(c)** Captures two corners

Figure 1: Y-structures that capture (a) the top corner or (b), (c) both top and right corners.

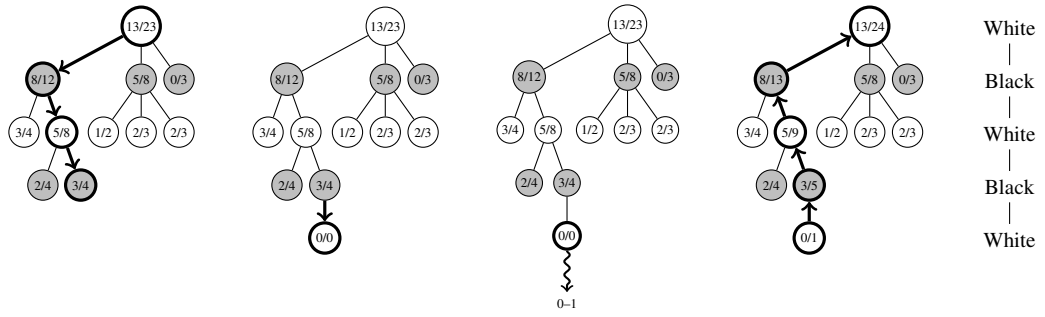


Figure 2: Monte-Carlo Tree Search. Nodes are labelled with the number of wins/playouts for White.

Poly-Y is played between two players, which we call *White* and *Black*. Players alternatingly place stones in empty cells, starting with White. The goal of the game is to capture the majority of the corners. A corner is captured by creating a Y-structure (examples are shown in Figure 1), which is a connected group of stones that (1) connects two adjacent edges, and (2) connects to any other edge on the board. The second rule ensures that once a corner has been captured by some player, it remains captured by that player. Since a full board has all corners captured and the board has an odd number of corners, no game can end in a draw.

As is the case in Hex, in practice, White has a strong first-player advantage. To mitigate this, there is a *swap rule* stating that Black may choose to swap sides as its first move. That is, the first move of White becomes the first move of Black. The *swap rule* results in more balanced games in practice: White will neither open with a very strong move, since then Black will swap, nor a very weak move, which allows Black to win easily.

Our player is based on the Monte-Carlo tree search (MCTS) algorithm [8], which has already been very successful in Hex and Go. We leverage work on MCTS Hex players to construct an MCTS player for Poly-Y, for which we describe the details in Section 2. MCTS uses random playouts in self play to evaluate the strength of game states. Guiding these random playouts toward more realistic games can improve the quality of an MCTS player. In Section 3, we discuss how we do random playouts, and we discuss a number of heuristics for Poly-Y to guide these playouts. In Hex it has been found that MCTS is weak in the very early stages of the game. To improve the performance of our player at these early stages, we have constructed an opening book, as we discuss in Section 4. Finally, in Section 5 we show the results of experiments performed on our Poly-Y player to show the effectiveness of our techniques.

2 Leveraging MCTS in Hex to Poly-Y

Others have already explored MCTS and strategies for Hex [2]. We leverage these strategies and adapt them to a Poly-Y player. Compared to the Hex players, which usually have 5 to 30 minutes of thinking time per game, our player only has 30 seconds of computation time. With this in mind, we have chosen to only use lightweight techniques.

Pure Monte-Carlo Search Traditional minimax search does not lead to strong play in the game of Hex, because of the high branching factor, and the difficulty of evaluating the strength of game states. Monte-Carlo search is an alternative method for evaluating the strength of game states. The idea of pure Monte-Carlo search is to compare the relative strength of moves in self play by measuring the average outcome of a number of random playouts from the states corresponding to these moves.

Monte-Carlo Tree Search As the basis for our player, we use the Monte-Carlo tree search algorithm (MCTS) [8], which has already been very successful in the game of Hex. The idea of MCTS is to combine pure Monte-Carlo search with a game tree similar to minimax. Like in minimax, in MCTS a game tree is constructed where nodes correspond to game states, and edges correspond to moves. Each node stores the total number of playouts performed from this node, and the number of winning playouts.

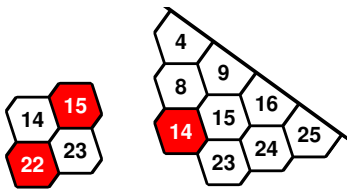
The MCTS algorithm works in four steps. The first step (Figure 2a) selects a promising node to be explored. Selection recursively selects the move with the highest *score*, which is computed based on statistics kept in the node. One challenge in the MCTS algorithm is to find a good scoring function that balances between exploration of moves that have few samples, and exploitation of the best moves found so far. This problem is an instance of the multi-armed bandit problem [3]. A common solution is to use the UCT (Upper Confidence Bound 1 applied to trees) algorithm [12] to score nodes. However, we use the all-moves-as-first heuristic to score nodes instead, as discussed in the next paragraph. After selecting a move to be explored for which no node exists, the tree is expanded (Figure 2b) by creating a leaf node, from which a Monte-Carlo playout (Figure 2c) is performed. To reduce the overhead of MCTS, our player performs 32 playouts when creating a new node. Finally, the result of the playout is propagated back up to the root (Figure 2d), while the statistics of all nodes on the way are updated.

All Moves as First If we consider a random playout, we alternatingly fill the board with random moves for both players. The order of these moves does not matter for the end results. The idea of the all-moves-as-first (AMAF) heuristic [5] is that any of these moves could have been the first move, and we can use this observation to gather information about many moves from a single playout. In effect, AMAF increases the number of playouts that we can perform by the number of empty cells on the board.

We now briefly review the AMAF heuristic. We say that a playout is made *from node* n if the playout was made in n or in any of its descendants. The *move that corresponds to node* n is the move performed to reach n from its parent. In the AMAF algorithm, each node n in the tree maintains two separate (wins, samples)-pairs: a direct pair (d_w, d_s) and an indirect pair (i_w, i_s). The *direct* pair maintains the cumulative wins d_w and samples d_s of all playouts made from n. The *indirect* pair maintains the cumulative wins i_w and samples i_s of all playouts made from any *sibling* of n where the move corresponding to n was played in the playout by the player who's turn it is. In the MCTS selection phase, we compute the *direct win rate* of a node as $w_d = d_w/d_s$, and the *indirect win rate* as $w_i = i_w/i_s$. Note that the indirect win rate w_i is based on playouts where the move was played in a different context, and is less reliable than the direct win rate w_d. We compute the score of a node using alpha-AMAF [10] as $w = \alpha \times w_d + (1 - \alpha) \times w_i$. For our player, we empirically found that $\alpha = 0.75$ produces the best results. Alternatively, the RAVE heuristic can be used, which increases α as more direct samples become available. However, we found that RAVE is not beneficial with the low number of playouts that we can produce, which corresponds to the findings of Helmbold and Parker-Wood [10].

One benefit of AMAF is that we can quickly explore moves. Like in MCTS Hex [2], we found no need for UCT exploration anymore, instead relying completely on AMAF for the exploration. Moreover, we found that AMAF quickly removes inferior moves.

Heuristics In Hex, patterns have been discovered that simplify the analysis of the game. One of these patterns is a bridge, as shown in Figure 3a. No matter what the opponent does, we can always create a connection between cells 15 and 22. If the opponent plays cell 14, we play cell 23 to create a connection, and vice versa. Cells 15 and 22 are said to be *virtually connected*. Another example of a virtual connection is shown in Figure 3b, where cell 14 is virtually connected to the edge of the board. These virtual connections also apply to Poly-Y.



(a) Bridge **(b)** Level-2 connection

Figure 3: Patterns that guarantee a connection (a) between stones and (b) to side.

In MCTS Hex players, virtual connections are used in two ways. First, virtual-connection search can be used to solve the game board [9]. However, this search is PSPACE-complete [11], and is therefore not feasible in limited time. H-search is a weaker form of virtual-connection search [1], and has been successfully applied in analyzing game positions for MCTS Hex [2]. However, H-search is still computationally expensive, which makes it infeasible for our player, because we have only 30 seconds per game. Second, virtual connections can be used in an even weaker form, by recognizing basic patterns on the board, and applying fixed responses. This technique is used in MCTS Hex to increase the quality of playouts by defending basic virtual connections [2]. We also use patterns to increase the quality of playouts in Poly-Y, as discussed in section 3.

In Hex, a board can be analyzed for inferior and dominating moves, which are provably irrelevant for the remainder of the game [2]. This technique can also be used for Poly-Y. For instance, playing

at the edge is generally weak, unless a (virtual) connection to the edge is made by this move, or when preventing the opponent from making a (virtual) connection. For our player, we found that the AMAF heuristic is already very effective at pruning weak moves, so we do not perform explicit inferior and dominating move analysis.

Win Condition For our win check, we use a flood-fill algorithm to find connected edges and captured corners. An alternative method is bitwise-parallel reduction (BPR). For the game of Y, BPR can speed up the win check by a factor of up to 15 compared to flood fill (Path) [4]. For Hex, this speedup is lower, because a Hex board has to be encoded into a larger Y board. Similar to Hex, we can encode a Poly-Y $(5, r)$-board into a Y board of height $3 \times r$. However, BPR only checks for a single Y-structure, therefore our encoding can only detect Y-structures's between two edges around a specific corner, so the reduction must be applied to five rotations of the board. Together with the encoding overhead, the expected performance of BPR is similar to flood fill. Thus, BPR is not helpful for Poly-Y and the simpler flood fill is the algorithm of choice.

3 Monte-Carlo Playouts for Poly-Y

The effectiveness of the Monte-Carlo method depends both on the quantity and the quality of playouts. Completing games in a purely random fashion does not correspond to the way players react in reality. For example, if a vital virtual connection is attacked, a rational player will defend this connection. This problem has already been observed in MCTS Hex [2], where heuristics are used to guide the Monte-Carlo search towards more realistic games. The idea of these heuristics is to maintain virtual connections. However, seemingly good heuristics do not necessarily improve gameplay, as shown in MCTS Hex players [2]. In this section, we describe the heuristics that we use for Poly-Y.

Defending Bridges As in MCTS Hex, we try to defend any bridge that is attacked by the opponent during a playout. In practice, a bridge may not be required for a win, and defending such a bridge wastes a move. We do not check for this in the playout, as this is too expensive. Instead, we rely on MCTS to avoid choosing such moves, since the tree search also considers moves that do not defend the bridge.

Higher-level Virtual Connections In MCTS Hex, patterns are also used for level-2 (Figure 3b) and level-3 virtual connections [2]. However, these patterns are quite large, and therefore slow to apply. Instead, we only check if the opponent tries to block us from reaching an edge, and avoid this by 'walking around the opponent' towards the edge. This pattern is computationally cheap to check, but is not exact in applying the level-2 and level-3 patterns. However, we found that for our player this simple pattern produces stronger play than more accurate level-2 and level-3 virtual connection patterns.

Fillboard: Avoid Playing at the Edges In purely random games, the order of moves does not matter. However, in the presence of heuristics, the order of moves affects the applicability of the heuristics. The fillboard heuristic is used in the MCTS Go program MoGo [6]. The idea of this heuristic is to distribute moves evenly over the board early in the game, resulting in a structure that can be exploited by other heuristics. We use a similar heuristic: not playing at the edges of the board early in the playout. The idea of this heuristic is that cells at the edges of the board are mostly filled in to defend virtual connections, and are usually bad moves if not part of a virtual connection. In our implementation, no randomly selected move will be at the edge of the board within the first 50 moves of the playout.

Implementation Performing more Monte-Carlo playouts leads to a stronger player, so our implementation should be as fast as possible. Therefore, we need to be able to apply the heuristics quickly. We only check for heuristic patterns around the last opponent move, and see if we need to respond to this move. To do this we compile our heuristics into a list of templates for each board position, which we can check in constant time each. We represent the board as two bit arrays, where each array corresponds to the stones of one of the players. A pattern consists of bitmasks that describe the state of the board, together with a move to play. Using bitwise operations, the bitmasks can be checked in constant time against the game state to see if the pattern matches. If multiple patterns match, we randomly pick one of the matching patterns. If no pattern matches, we pick a random move.

4 Opening Analysis

We found that the outcome of a game is usually determined in the first few moves of a game. For example, we have seen games where after 10 moves the game was already determined. However, Monte-Carlo tree search does not perform very well in the beginning of the game. To improve the performance of our player early in the game, we have constructed an opening book for a Poly-Y $(5, 7)$-board.

Methodology Computing a perfect opening book is computationally infeasible. Instead, we compute an opening book by evaluating the strength of game states after an opening sequence by measuring the win / loss ratio from this state in self play. Assuming that our opponents also use a variant of MCTS, we expect that the measured win rates are a good representative of our actual chance of winning. When we evaluate states that are deeper in the game tree, we also get more information about the strength of earlier moves. However, the number of opening sequences increases exponentially in the length of the opening sequence. Furthermore, it is computationally expensive to evaluate the strength of a game state. In our opening analysis, we use 5 seconds per player for the self plays, for a total of 10 seconds per game. We perform 256 self plays per state, which means that evaluating a single opening sequence takes 42.7 minutes of single-core computation time. Still, the variation in measurements is high, as the error decreases only by the square root in the number of self plays. Because the board is symmetrical, we only have to compute opening moves for one symmetry, so there are only 16 opening moves to consider. It takes 11.4 hours of single-core computation time to evaluate these 16 opening moves. If we want to look deeper, we have to consider all 105 responses the opponent can make. For this level-2 analysis, there are 1680 cases to consider, for a total of 49.8 days of single-core computation time. A level-3 analysis would require analyzing 174,720 states, for a total of 14.2 years of single core computation time. This shows that computing a deep opening book is infeasible. An optimization is to only investigate the most promising moves. For instance, we could stop analyzing a branch when our win rate in self play is higher than some threshold. Alternatively, a method such as UCT can be used to perform fewer playouts for game states that are less promising, and instead focus on the more promising game states [7]. An additional optimization is to prune states that cannot become better than the best state found so far with more samples, e.g., if it is outside the 95% error bounds of the best move. In our analysis we do not use these optimizations methodically, but we hand picked the best moves for further exploration.

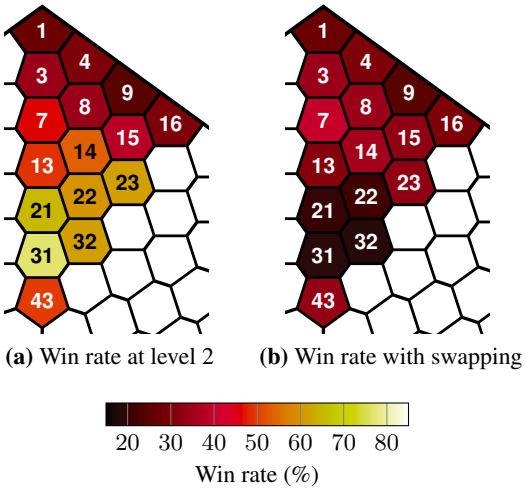
Results We have performed our analysis on a 64-core (4×16-core 2.0 GHz) AMD machine by running 64 independent games in parallel, which allows us to perform the level-2 analysis in 18.7 hours. After performing the level-2 analysis, we decided on our opening move, and we decided for which opponent opening moves we would use the swap rule. For the moves where we use the swap rule, we computed the best level-3 responses for the top level-2 moves, which we can play if the opponent plays one of these top level-2 moves. For the moves where we do not swap, we did not perform any further computation. In case the opponent chooses to swap on our opening move, we can play the best level-2 move. Additionally, for the top level-3 moves we have also computed the best level-4 response, which we can use if we are Black. In total, we used two weeks of computation time to compute our opening book. The resulting opening book can be downloaded from our source repository[4]. Table 1 shows the win rate for White after analyzing the opening move (level 1), after analyzing the best response from Black (level 2), and after analyzing the best response for White (level 3). Figure 4a graphically depicts the win rates for White after the first move of Black, assuming Black does not swap. Figure 4b depicts the win rates for White either after the second move of Black when his first move was swap, or after the first move of Black otherwise.

The win rates we found varied wildly when analyzing the positions deeper. For instance, move 15 has a win rate of 49.6% at level 1, while at level 2 the win rate drops to 37.9%, and at level 3 it is back up to 67.6%. However, note that due to the way the opening book is constructed, these win rates do correspond to actual playing strength in self play against a version of our player without an opening book. To select a good opening move, we also have to consider the swap rule. Ideally, we want a move that leads to a 50% win rate at the deepest examined level. As we did not find such a move, we picked move 15 with a win rate hovering around 50%. Alternatively, moves 8 and 3 seem good options as well.

[4] Opening book: `https://github.com/lwevers/lynx/`

Opening	Level 1	Level 2	Level 3	Swap?
9	40.8%	24.6%	-	
1	39.4%	27.3%	-	
4	38.8%	30.0%	-	
16	40.2%	30.0%	-	
8	45.4%	33.6%	-	
3	46.2%	33.6%	-	
7	59.4%	46.5%	60.5%	✓
14	58.0%	53.1%	64.8%	✓
43	66.5%	49.6%	66.8%	✓
15	49.6%	37.9%	67.6%	✓
23	63.7%	61.3%	67.6%	✓
13	65.0%	48.8%	69.1%	✓
22	64.9%	59.7%	78.5%	✓
21	67.2%	64.8%	80.0%	✓
31	79.4%	76.6%	81.3%	✓
32	76.7%	61.3%	81.6%	✓

Table 1: Win rate for all opening moves.



(a) Win rate at level 2 **(b)** Win rate with swapping

Figure 4: Win rate (a) without and (b) with swap rule.

We also use the results of the opening analysis to determine when to apply the swap rule. If the opponent opens with a move that has less than 50% win chance for us at the deepest level that we analyzed, we apply the swap rule.

5 Experimental Results

We have performed experiments with our player in self play, which we discuss in this section. In the experiments, our player alternatingly plays Black and White, except for the opening book experiment. Our player uses $\frac{\text{remaining time}}{12}$ time per move. As a baseline, our player uses 20 seconds of computation time for the whole game, which corresponds to the amount of time that our player used in the CodeCup competition. In the results, intervals show 95% confidence bounds, computed with the Wilson score interval with continuity correction [13].

Game Duration We investigated the length of games from a sample of 256 games. The first games end by move 50, and by move 80 nearly all games have ended. Interestingly, games end earlier when our player plays without heuristics. When using heuristics our player gets to a winning state earlier, but the player does not try to end the game immediately because it considers all moves as winning.

The Effect of Computation Time Figure 5 shows the effect of computation time on the playing strength of our player. Playing strength is measured by the win rate against the baseline player. Additionally, the ELO gain or loss compared to the baseline is shown. We see that time has a large effect on playing strength. We can gain or lose this speed by the choice of algorithm, or the programming language implementation. We also see that spending more time leads to diminishing returns. This shows that it may be worthwhile to spend more time on heuristics and higher level analysis instead of playouts if more time is available.

Comparing Heuristics Figure 6 shows how various versions of our player compare to each other. Adding either AMAF or playout heuristics provides a significant improvement over a UCT version of our player. The AMAF version is better against the UCT version than the UCT with heuristics version. Interestingly, the UCT with heuristics version beats AMAF, showing that the win rates of different improvements cannot simply be multiplied to obtain the win rate when combining the improvements. This could be explained by AMAF effectively increasing the number of playouts, while heuristics lead a more targeted search in the tree. Combining both AMAF and playout heuristics leads to another large

Time	Win rate	ELO
1.25 s	9.6 % ± 2.0%	-390
2.5 s	15.7 % ± 2.4%	-292
5 s	27.9 % ± 2.9%	-165
10 s	38.0 % ± 3.1%	-85
20 s	51.7 % ± 3.1%	12
40 s	61.9 % ± 3.1%	84
80 s	68.3 % ± 3.0%	133
160 s	77.6 % ± 2.7%	216
320 s	79.5 % ± 2.6%	235

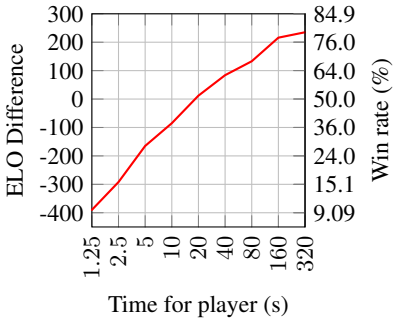


Figure 5: Win rate against 20 s (1024 playouts).

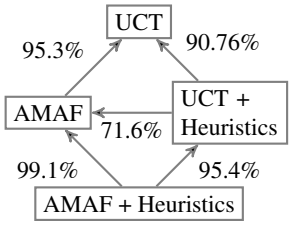


Figure 6: Win rates of various players (20 s, 10,000 playouts, ± 1% with 95% confidence).

Time per player	**5 s** (n = 20,000)	**20 s** (n = 20,000)	**80 s** (n = 10,000)
Bridges	97.5% ± 0.23%	98.8% ± 0.16%	99.2% ± 0.21%
Higher-level	67.3% ± 0.66%	70.4% ± 0.64%	72.4% ± 0.89%
Fillboard	57.4% ± 0.69%	58.7% ± 0.69%	58.3% ± 0.97%

Table 2: Incremental improvements of individual playout heuristics, with n playouts.

improvement over both the AMAF and the UCT heuristics version. In our experiments, the UCT-only version was able to beat the AMAF + Heuristics version only five times out of 10,000 games.

Table 2 shows the incremental improvements for the individual playout heuristics compared to the *previous entry in the table*, where the bridges heuristic is compared to a baseline featuring AMAF without heuristics. That is, the higher-level version also contains the bridges heuristic, and the fillboard version also contains the higher-level and bridges heuristics.

We see that the bridges heuristic provides a dramatic increase in playing strength. The heuristic to move around the opponent (higher-level) also increases playing strength by a large amount, and the fillboard heuristic provides another small increase in playing strength. We also see that, as more time is used and more playouts are performed, the performance of the heuristics improve.

Opening Book Table 3 shows the effect of the opening book on playing strength when playing as White or Black. The no book version of our player does not use the opening book, but opens with move 15 and swaps according to Table 1. Without an opening book, we see that Black has a strong first-player advantage due to the swap rule. When we play using the opening book against a version that does not use the opening book, we see a large increase in playing strength as White. However, playing strength of Black does not increase significantly. We also see that with more computation time, White has less of an advantage from using the opening book. Finally, when both players use the opening book, White gains the advantage over Black.

6 Conclusions

Many techniques used in MCTS Hex are applicable to Poly-Y. However, our player is provided with only 30 seconds of time per game, which makes computationally expensive techniques such as virtual-connection search infeasible. Moreover, we did not implement techniques for pruning moves, such as dominating move analysis, since early experiments showed that the AMAF heuristic effectively covers

Time per player	**5 s**		**20 s**		**80 s**	
Playing as	**White**	**Black**	**White**	**Black**	**White**	**Black**
No book vs. no book	25.3%	74.7%	24.4%	75.6%	23.2%	76.8%
Book vs. no book	69.0%	76.3%	65.1%	78.0%	62.2%	77.0%
Book vs. book	56.9%	43.1%	57.4%	42.6%	54.5%	45.5%

Table 3: Effect of using the opening book (10,000 playouts, ± 1% with 95% confidence).

this. Instead, we rely on tree search together with patterns for protecting virtual connections in the random playouts. Additionally, we found that detailed level-2 and level-3 patterns in the random playouts, as used by some Hex players, are less effective than a simpler 'walking to the edge' strategy. Finally, to improve the performance of our player in the early game, we computed an opening book, which made our player significantly stronger against our player without an opening book.

Future Work Our player only uses three simple playout heuristics, but these improve playing performance by a lot. Future work may investigate whether other patterns improve playing performance. Additionally, given more computation time per move, our player may improve through virtual-connection search as used in MCTS Hex players. It would also be interesting to investigate very lightweight virtual-connection search techniques that can be used in very short games. Recently, Saffidine and Cazenave [14] show that for the game Y, product propagation is an alternative to MCTS in settings with tight time constraints, so it would be interesting to investigate how this performs for our Poly-Y player.

References

[1] Vadim V. Anshelevich. A hierarchical approach to computer Hex. *Artificial Intelligence.*, 134(1-2):101–120, January 2002.

[2] Broderick Arneson, Ryan B. Hayward, and Philip Henderson. Monte Carlo tree search in Hex. *IEEE Trans. on Computational Intelligence and AI in Games*, 2(4):251–258, December 2010.

[3] Donald A. Berry and Bert Fristedt. *Bandit Problems: Sequential Allocation of Experiments*. Springer, 1985.

[4] Cameron Browne and Stephen Tavener. Bitwise-parallel reduction for connection tests. *IEEE Transactions on Computational Intelligence and AI in Games*, 4(2):112–119, June 2012.

[5] Bernd Brügmann. Monte Carlo Go. Technical report, Physics Department, Syracuse University, March 1993.

[6] Guillaume Chaslot, Christophe Fiter, Jean-Baptiste Hoock, Arpad Rimmel, and Olivier Teytaud. Adding expert knowledge and exploration in Monte-Carlo tree search. In *Advances in Computer Games*, volume 6048 of *LNCS*, pages 1–13. Springer, 2010.

[7] Guillaume Chaslot, Jean-Baptiste Hoock, Julien Perez, Arpad Rimmel, Olivier Teytaud, and Mark Winands. Meta Monte-Carlo tree search for automatic opening book generation. *Proceedings of the IJCAI'09 Workshop on General Intelligence in Game Playing Agents*, pages 7–12, 2009.

[8] Rémi Coulom. Efficient selectivity and backup operators in Monte-Carlo tree search. In *5th Int. Conference on Computers and Games*, volume 4630 of *LNCS*, pages 72–83. Springer, 2007.

[9] Ryan Hayward, Yngvi Björnsson, Michael Johanson, Morgan Kan, Nathan Po, and Jack van Rijswijck. Solving 7×7 Hex with domination, fill-in, and virtual connections. *Theoretical Computer Science*, 349(2):123–139, 2005.

[10] David P. Helmbold and Aleatha Parker-Wood. All-Moves-As-First Heuristics in Monte-Carlo Go. *Proc. of the 2009 International Conference on AI*, pages 605–610, 2009.

[11] Stefan Kiefer. Die Menge der virtuellen Verbindungen im Spiel Hex ist PSPACE-vollständig. Studienarbeit Nr. 1887, Universität Stuttgart, July 2003. In German.

[12] Levente Kocsis and Csaba Szepesvári. Bandit based Monte-Carlo planning. In *Proc. 17th European Conference on Machine Learning*, ECML'06, pages 282–293. Springer, 2006.

[13] Robert G. Newcombe. Two-sided confidence intervals for the single proportion: comparison of seven methods. In *Statistics in Medicine*, volume 17, pages 857–872, 1998.

[14] Abdallah Saffidine and Tristan Cazenave. Developments on product propagation. In *8th International Conference on Computers and Games*, volume 8427 of *LNCS*, pages 100–110. Springer, 2014.

Part II

Compressed papers (B)

Fair-Share ILS: A Simple State of the Art Iterated Local Search Hyperheuristic.[1]

Steven Adriaensen Tim Brys Ann Nowé

Vrije Universiteit Brussel, Pleinlaan 2 1050 Elsene, Belgium

Abstract

In this paper we present Fair-Share Iterated Local Search, a simple, state-of-the-art, hyperheuristic method. As a selection hyperheuristic it is an *off-the-peg* solution, and therefore readily applicable to new domains. Its simplicity and modularity, make it an interesting starting point for further research. In our experiments we analyse FS-ILS and show it to be competitive with the state-of-the-art.

1 Introduction

Most research in metaheuristics is focused on problem-specific techniques. As a consequence, metaheuristic methods are not readily applied to newly encountered problems, or even new instances of similar problems. Furthermore, the cost of developing and applying *made-to-measure* metaheuristic solutions is high, resulting in few practical applications.

Recently, there is a renewed interest in general metaheuristic methods. Rather than attempting to outperform *made-to-measure* methods, these methods provide a cheap *off-the-peg* alternative. A popular approach in this renaissance are hyperheuristics [3]. A hyperheuristic combines a given set of low level heuristics to solve a given problem instance, in a problem independent way. Here, low level heuristics are construction, perturbation and recombination heuristics used in the problem instance domain. They can be applied to any domain, provided a set of low level heuristics is defined for it first.

In this paper [2], we present Fair-Share Iterated Local Search (*FS-ILS*), a selection hyperheuristic. *FS-ILS* was obtained in [1] using a semi-automated design approach and is designed to be reusable in research and by practitioners. In its implementation and analysis we've used the Hyflex framework [4]. This framework was used to support the CHeSC (2011) competition and currently provides benchmark instances and problem-specific components for 6 different combinatorial optimization problems.

2 Method

In this section we describe the Fair-Share Iterated Local Search method. Algorithm 1 shows the high level search strategy of *FS-ILS*. Next we describe the different aspects of the algorithm in more detail.

Algorithm 1 High level search strategy of Fair-Share Iterated Local Search

1: $c_{current} \leftarrow$ constructSolution()
2: **while** $t_{elapsed} < t_{allowed}$ **do**
3: *Select a non-greedy perturbation heuristic* llh_{pert}
4: $c_{proposed} \leftarrow llh_{pert}(c_{current})$
5: $c_{proposed} \leftarrow$ localSearch($c_{proposed}$)
6: **if** accept($c_{proposed}$) **then** $c_{current} \leftarrow c_{proposed}$
7: **if** restart() **then** $c_{current} \leftarrow$ constructSolution()
8: **end while**

[1] The full paper has been published in *Proceedings of the 16th annual conference on Genetic and evolutionary computation* [2].

Selection heuristic (line 3): Selects a domain-specific non-greedy perturbation heuristic proportionally to the number of new incumbent solutions (accepted and new proposals) it generated per time unit it was used ($\frac{\text{\# new incumbent candidate solutions generated by } h}{\text{total time spent on } h}$), in previous applications.

Iterative improvement (line 5): Iteratively applies a random *greedy* domain-specific perturbation heuristic. When its application does not lead to improvement, the heuristic is excluded, until some other heuristic finds improvement. If all heuristics are excluded, local search is terminated.

Acceptance condition (line 6): Accepts $c_{proposed}$ with a probability $e^{\frac{f(c_{current})-f(c_{proposed})}{T*\mu_{impr}}}$, where f is the domain-specific evaluation function and μ_{impr} the average improvement in previous improving iterations. The temperature T is a positive constant and a parameter of the algorithm (default 0.5).

Restart condition (line 7): Restarts the search when $w > \frac{t_{allowed}}{t_{elapsed}} * w_{max}$, where w is the number of iterations passed since obtaining the best candidate solution since last restart and w_{max} the greatest value of w ever observed. As an exception, the algorithm is not restarted when the time remaining is less than the time it took to find a solution as good as the best candidate solution obtained so far.

3 Experiments

In this section we briefly summarize our experimental results. First, we performed a parameter sensitivity analysis, showing that *FL-ILS* is largely robust to changes in its sole parameter (T). Subsequently we analyzed the presence of accidental complexity, showing that *FS-ILS* performs significantly better than simpler variants, as such motivating every design choice. Finally, we compared *FS-ILS* to the state-of-the-art. We compared *FS-ILS* to the contestants of the CHeSC 2011 competition and found that *FS-ILS* would have won the competition and performs best on 3 out of the 6 domains. In addition we compared the median performance of *FS-ILS* to that of *AdapHH*, the winner of the CHeSC competition, on all 68 HyFlex benchmark instances. We observe that *FS-ILS* performs (significantly) better than *AdapHH* on more instances than vice versa. This for both the instances used, and not used, during the competition. These results illustrate *FS-ILS*'s generality and competitiveness on this challenging benchmark set.

4 Conclusion

In this paper, we presented *FS-ILS*, a simple, state-of-the-art, Iterated Local Search, selection hyperheuristic, discovered in prior research, using a semi-automated design approach [1].

There are a large number of metaheuristic methods. What is the value of yet another method? As a hyperheuristic, *FL-ILS* is an *off-the-peg* solution, and therefore readily applicable to new domains. Unlike other state-of-the-art methods, it is rather simple and therefore understandable and reproducible.[2] These properties make *FS-ILS* an interesting starting point for future hyperheuristic research.

References

[1] Steven Adriaensen, Tim Brys, and Ann Nowé. Designing reusable metaheuristic methods: A semi-automated approach. In *Evolutionary Computation (CEC), 2014 IEEE Congress on*. IEEE, 2014.

[2] Steven Adriaensen, Tim Brys, and Ann Nowé. Fair-share ils: A simple state-of-the-art iterated local search hyperheuristic. In *Proceedings of the 16th annual conference on Genetic and evolutionary computation (GECCO)*. ACM, 2014.

[3] Edmund K Burke, Matthew Hyde, Graham Kendall, Gabriela Ochoa, Ender Ozcan, and Rong Qu. Hyper-heuristics: A survey of the state of the art. *Journal of the Operational Research Society*, 64(12):1695–1724, 2013.

[4] Gabriela Ochoa, Matthew Hyde, Tim Curtois, Jose Vazquez-Rodriguez, James Walker, Michel Gendreau, Graham Kendall, Barry McCollum, Andrew Parkes, Sanja Petrovic, et al. Hyflex: a benchmark framework for cross-domain heuristic search. *Evolutionary Computation in Combinatorial Optimization*, pages 136–147, 2012.

[2] Code can be found at: https://github.com/Steven-Adriaensen/FS-ILS.

On the Input/Output Behavior of Argumentation Frameworks[1]

Pietro Baroni [a] Guido Boella [b] Federico Cerutti [c]

Massimiliano Giacomin [a] Leendert van der Torre [d] Serena Villata [e]

[a] *Dipartimento di Ingegneria dell'Informazione, University of Brescia, Italy*
[b] *Dipartimento di Informatica - University of Torino, Italy*
[c] *Department of Computing Science - University of Aberdeen, UK*
[d] *Compurer Science and Communication, University of Luxembourg*
[e] *INRIA Sophia Antipolis - Mediterranee, France*

This paper deals with modularity in abstract argumentation. The "Merriam-Webster Learner's Dictionary" defines *modular* as "having parts that can be connected or combined in different ways" while the "Free Dictionary online" remarks that modularity is intended "for easy assembly and repair or flexible arrangement and use". Modularity is a highly desirable property, often enforced by design, in any kind of either material (like the popular Lego toys) or immaterial (like programs developed according to the object-oriented paradigm) artifacts, including knowledge representation and reasoning formalisms.

Roughly speaking, modularity involves two main properties, namely *separability* and *interchangeability* of modules. As to the former, it has to be possible to describe and analyse the global behavior of an artifact in terms of the combination of the local behaviors of the modules composing it. Each local behavior can be characterized individually in a way which is independent of the internal details of the other modules (and, in a sense, of the module itself) and captures only the connections and mutual interactions between the module and the other ones. In other words, each module can be described as a black-box whose Input/Output behavior fully determines its role in the global behavior of any artifact it is plugged in. As to the latter, the interest in replacing a module with another one is very common and arises from a large variety of motivations, either at the operational or design level. Interchangeability of two modules requires first of all that they are compatible as far as the connections with the rest of the artifact are concerned, i.e. that the interfaces they expose are such that wherever one of the modules can be "plugged in", the other can too. Besides this *plug-level* interchangeability, it is of great interest to characterize the *behavior-level* interchangeability of modules, namely to identify the situations where internally different modules can be freely interchanged without affecting the global behavior of the artifact they belong to, since their Input/Output behavior is equivalent in this respect.

While the formalism of abstract argumentation frameworks and the relevant argumentation semantics do not appear to have been designed with modularity in mind, investigating their relevant properties is an important research topic which, after having been somehow overlooked, is attracting increasing attention in recent years. An argumentation framework is basically a directed graph representing the conflicts between a set of arguments (the nodes of the graph) and an argumentation semantics can be regarded as a method to answer (typically in a non univocal way, i.e. producing a set of alternative answers) the "justification question": "Which is the justification status of arguments given the conflict?"

Referring to a representative set of semantics proposed in the literature, (namely admissible, complete, grounded, preferred, stable, semi-stable and ideal semantics) this paper provides a systematic and comprehensive assessment of modularity in abstract argumentation, by identifying and analyzing in this context the formal counterparts of the general notions of separability and interchangeability.

[1] Appeared as: Pietro Baroni, Guido Boella, Federico Cerutti, Massimiliano Giacomin, Leendert van der Torre, Serena Villata: On the Input/Output behavior of argumentation frameworks. Artificial Intelligence 217: 144-197 (2014).

Given a partition of an argumentation framework into *partial* (or *local*) interacting subframeworks, analyzing separability consists in addressing the following issues:

- "Is it possible to define a local counterpart of the notion of semantics?" i.e. "Is there a method to produce local answers to the justification question, taking into account the interactions with other subframeworks?"
- "Can the set of justification answers prescribed by the (global) semantics be obtained by properly combining (in a bottom-up fashion) the sets of local answers produced in the subframeworks by its local counterpart?"
- symmetrically, "Can the sets of local answers be obtained (in a top-down fashion) as projections onto the subframeworks of the global answers?"

As to the first issue, we introduce the notion of *local function* for a subframework and show that under very mild requirements, satisfied by all semantics considered in this paper, it is possible (and easy) to identify the *canonical local function* for a global semantics. As to the second and third issues, we introduce the formal notions of top-down and bottom-up decomposability, which, jointly, correspond to the notion of (full) decomposability of an argumentation semantics.

Strong as it may seem, full decomposability with respect to every arbitrary partition of every argumentation framework is not unattainable. Indeed, we show that it is satisfied by some of the semantics considered in this paper, while some others are able to achieve at least top-down decomposability and the remaining ones lack all decomposability properties.

As arbitrary partitions correspond to a completely free (if not anarchical) notion of modularity, we also consider a "tidier" style of partitioning, involving the graph-theoretical notion of *strongly connected components*. It turns out that, restricting the set of partitions this way, helps some, but not all, semantics to recover full decomposability.

Turning to interchangeability, we deal with both its plug-level and behavior-level aspects. As to the plug-level, borrowing some terminology from circuit theory, we introduce the notion of *argumentation multipole* as a generic replaceable argumentation component, namely a partial framework interacting through an input and output relation with an external set of invariant arguments.

Plug-level compatibility of two multipoles is a very relaxed notion, since it is only required that two multipoles refer to the same set of external arguments. This is motivated by the fact that imposing a tighter correspondence between Input/Ouput "terminals" of the multipoles would unnecessarily restrict the scope of the subsequent analysis on behavior-level compatibility. In fact, our analysis shows that a sensible notion of behavioral equivalence between multipoles (called *Input/Output equivalence*) can be introduced by requiring that the effect of the multipoles on the external arguments is the same: it may well be the case that multipoles with different "terminals" have the same effect in behavioral terms. Of course, Input/Output equivalence is a semantics-dependent notion since the behavior of a multipole can only be defined by referring to a specific semantics using the notion of local function mentioned above. In particular, it may be the case that two multipoles are equivalent with respect to some semantics and not equivalent with respect to another semantics.

Input/Output equivalence is the basis for the analysis of the operation of replacement within an argumentation framework. Basically, a replacement consists in substituting a part of the framework with a plug-level compatible multipole. While this notion *per se* allows for arbitrary substitutions, one is interested in analysing those replacements which have a sound basis. In this perspective, building on multipole equivalence, it is possible to identify the semantics-dependent notions of *legitimate* and *contextually legitimate* replacement, the former being stronger than the latter since legitimate replacements are a (typically strict) subset of contextually legitimate replacements.

One might expect that, given a semantics, legitimate (with respect to that semantics) replacements ensure that the invariant part of the framework is unaffected (in a sense, that it does not notice the change). This property is called *semantics transparency*. A stronger expectation (since the requirement on the replacements is weaker) would be that the invariant part of the framework is unaffected for any contextually legitimate replacement: this property is called *strong transparency*.

Natural as it may seem, transparency is not achieved by all semantics and requires a detailed analysis, showing that different levels of transparency are achieved by the semantics considered in this paper, also taking into account different restrictions on the set of allowed replacements.

Simplifying the Visualization of Confusion Matrix

Emma Beauxis-Aussalet [a] Lynda Hardman [a]

[a] *CWI - Information Access Group - Science Park 123 - Amsterdam*

Abstract

Supervised Machine Learning techniques can automatically extract information from a variety of multimedia sources, e.g., image, text, sound, video. But it produces imperfect results since the multimedia content can be misinterpreted. Errors are commonly measured using confusion matrices, encoding type I and II errors for each class. Non-expert users encounter difficulties in understanding and using confusion matrices. They need to be read both column- and row-wise, which is tedious and error prone, and their technical concepts need explanations. Further, the visualizations commonly use of complex metrics, e.g., Precision/Recall, F1 scores. These can be overwhelming and misleading for non-experts since they may be inappropriate for specific use cases. For instance, type II errors (False Negative) are critical for medical diagnosis while type I errors (False Positive) are more tolerated. In the case of optical sorting of manufactured products (defect detection), the sensitivity to errors can be the opposite. We propose a novel visualization design that address the needs of non-experts users. Our visualization is intended to be easier to understand, and to minimize the risk of misinterpretation, and so for all kind of use cases. Future work will evaluate our design with both experts and non-experts, and compare its effectiveness with that of traditional ROC and Precision/Recall curves.

1 Use cases

Confusion matrices are the major mean to evaluate errors in classification problems. They encode the complete specification of misclassifications: the numbers of misclassified items for each pair {original class in which items should be classified, incorrect class in which items are erroneously classified}. Confusion matrices are used for: i) inspecting errors for each class; ii) tuning software parameters such as detection thresholds; iii) comparing software versions. The selection of software version, or parameter settings, basically rely on the tolerance to *Type I or II* errors. The sensitivity to either error type depends on application domains. In some domains, type I are critical while type II are more tolerated: e.g., fraud detection involving automatic suspension of services (bank, mail, social media), biometric identification, recommendation, optical sorting (*Case A*). In other domains, type II are critical while type I are more tolerated: medical diagnosis, threat detection (*Case B*). Others are sensitive to both error types: character recognition, monitoring of population dynamics in ecology (*Case C*).

Analyzing confusion matrices is complicated since FN for one class are FP for another. Users need to inspect the matrix both column-wise and row-wise, and they can forget cell values, or may read only columns or rows. Confusion matrices are usually synthesized by cumulating misclassifications into FP, FN, TP and TN for each class. But users can no longer distinguish which classes are likely to be confused with another. This is an issue in domains analyzing trends over time (e.g., population dynamics in ecology). E.g., an important increase of one class imply an increase of its FN, and can induce a deceiving increase of other classes. Confusion matrices are synthesized further by deriving advanced metrics, e.g., TP rate, FP rates, F1 scores. Non-experts may not know which metrics suit their use case, or misinterpret them. E.g., high TN may conceal critical errors by yielding low *FP Rate* and high *Accuracy*. *Precision* does not convey the errors critical for *Case A*, nor *Recall* and *FP Rate* for *Case B*, nor *Accuracy* and *F1 score* convey the errors critical for neither *Case A and B*. For *Case C*, using only one metrics amongst *Precision, Recall and FP Rate* does not convey sufficient information. ROC and Precision/Recall curves, increase the risk to overwhelm and confuse users. Non-experts may

not identify the point offering the best tradeoff between type I and II errors. Further, classifiers' errors can appear identical in one type of curve, or almost identical, while another view would reveal further differences (Fig. 1).

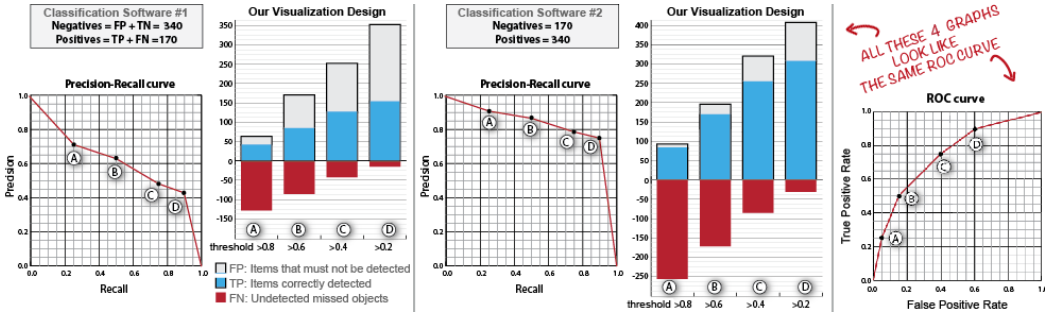


Figure 1: Alternative visualizations: our design, and equivalent ROC and Precision/Recall curves.

2 Visualization Design

To ease non-experts' interpretation, we display only TP, FN and FP (Fig. 1). TN are omitted since they are potentially uninteresting (e.g., not contained in end-results) and misleading (e.g., high TN yield high *Accuracy*). We primarily show raw numbers of errors (e.g., in Fig. 1) which are more tangible, without rates and advanced metrics. It preserves the numbers of items in the ground-truth, hidden in ROC and Precision/Recall curves. It displays both type I and II errors, without attempting to show rates relatively to case-dependent frames of reference. Hence it suits most use cases.

Classes can have heterogeneous numbers of ground-truth items, thus being difficult to compare. Hence our design in Fig. 2 reports errors proportionally to the TP of each class. It details inter-classes confusions by indicating the 2 classes producing most FP and FN, and groups errors for all other classes. It indicates potential biases in end-results (e.g., high confusion between classes yielding correlated but unrepresentative trends in end-results). Such information are lost with the usual metrics, since FN and FP are cumulated over classes.

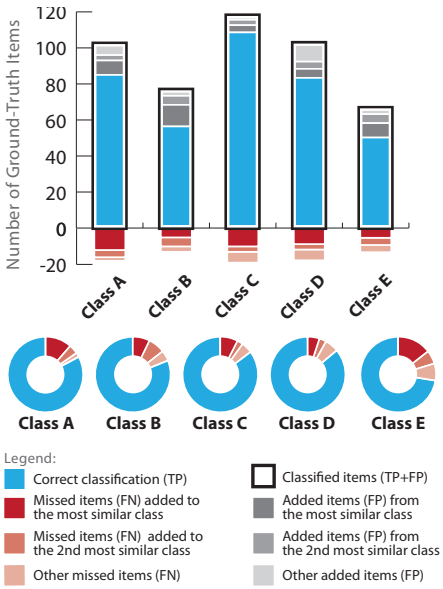


Figure 2: Visualization of inter-classes confusions.

Our design is of interest for Machine Learning researchers and practitioners dealing with the acceptance of their software by end-users. Novel applications of Machine Learning techniques face trust issues: users need to assess their reliability, while evaluations provided by experts are perceived as abstruse, thus impairing further user trust. Our visualization provides a solution for fully conveying the performance of Machine learning software, while minimizing user cognitive effort and remaining intuitive [1, 2]. It can also be used by Machine Learning experts themselves.

References

[1] E. Beauxis-Aussalet, E. Arslanova, L. Hardman, and J. Van Ossenbruggen. A case study of trust issues in scientific video collections. In *Proceedings of the 2nd ACM international workshop on Multimedia Analysis for Ecological Data*, 2013.

[2] E. Beauxis-Aussalet and L. Hardman. Visualization of confusion matrix for non-expert users. In *Proceedings of the IEEE Symposium on Information Visualization (IEEE InfoVis), in press*, 2014.

LOD Laundromat:
A Uniform Way of Publishing Other People's Dirty Data*

Wouter Beek [a] Laurens Rietveld [a] Hamid R. Bazoobandi [a]
Jan Wielemaker [a] Stefan Schlobach [a]

[a] *Dept. of Computer Science, VU University Amsterdam, NL*

Abstract

It is widely accepted that *proper* data publishing is difficult. In the realm of Linked Open Data (LOD) it is particularly important for data to be published correctly since data is intended to be processed by machines which cannot always come up with fallback procedures for data imperfections. This paper presents the LOD Laundromat which removes stains from data without any human intervention. This fully automated approach is able to make very large amounts of LOD more easily available for further processing. The LOD Laundromat provides researchers and application developers with a wealth of data that is guaranteed to conform to a specified set of best practices, thereby greatly improving the chance of data actually being (re)used.

1 Problem statement

Uptake of Linked Open Data (LOD) has seen a tremendous growth over the last decade or so. Due to the inherently heterogeneous nature of interlinked datasets that come from very different sources, LOD is not only a fertile environment for innovative data (re)use, but also for mistakes and incompatibilities [2]. Such mistakes include character encoding issues, socket errors, protocol errors, syntax errors, corrupted archive headers and authentication problems. Existing solutions for cleaning LOD (standards, guidelines, tools) are targeted towards human data creators, who can (and do) choose not to use them. Therefore, despite many efforts, much of LOD is still difficult to use today, mostly because of mistakes for which solutions exist. We believe that this poses an unnecessary impediment to the (re)use of LOD for academic and commercial purposes.

2 LOD Laundromat

We present the LOD Laundromat, a data cleaning system that takes *immediate* action by targeting the *data* directly, not its maintainers. By cleaning stains in LOD without any human intervention LOD Laundromat is able to make very large amounts of LOD more easily available for further processing *right now*. The collection of cleaned datasets that LOD Laundromat produces are standards- and guidelines-compliant siblings of existing, idiosyncratic datasets. The data-oriented approach of LOD Laundromat is complementary to existing efforts, since it is preferable that someday the original dataset is cleaned by its own maintainers. However, we believe that until that day, our complementary approach is necessary to make LOD succeed while the momentum is still there.

The LOD Laundromat is available at `http://lodlaundromat.org`. The collection of datasets that is comprises is continuously being extended. Anyone can add seed points to the LOD Laundry Basket[1] by eneringing URLs into a Web form. The fully automated LOD Washing Machine[2] takes seed points from the LOD Laundry Basket and cleans them. Cleaned datasets are disseminated in the LOD Wardrobe[3]. Metadata for each cleaned dataset is available via a public SPARQL enpoint[4].

*The full paper has been published in *Proceedings of the International Semantic Web Conference*, 2014.

[1] `http://lodlaundromat.org/laundryBasket.html`

[2] `https://github.com/LODLaundry/LOD-Washing-Machine`

[3] `http://lodlaundromat.org/wardrobe.html`

[4] `http://lodlaundromat.org/sparql.html`

3 Quality criteria

Data that is cleaned by the LOD Laundromat is ensured to adhere to the following quality criteria:

Easy grammar: LOD Laundromat takes care of the data preprocessing stage, allowing subsequent processing of the data in a safe way. This is guaranteed by a uniform data format that is easy to parse in an unambiguous way, even by non-RDF tools such as Pig, grep, sed, and the like.

Speed: LOD Laundromat delivers data that can be processed by other tools in a speedy way by solving the major parsing inefficiencies for LOD. E.g., inefficient serialization formats such as RDF/XML and RDFa are converted to a more optimal format. In addition, syntax errors, which necessitate the parser to come up with fallback options that may be expensive, are ensured to be absent.

Quantity: LOD Laundromat makes clean data available for a large number of data documents (tens of thousands) and triples (tens of billions), thereby covering a large part of the LOD Cloud.

Flexibility: LOD Laundromat makes it easy to combine data documents. E.g., appending multiple documents into a single one normally requires the costly process of blank node renaming. Data delivered by the LOD Laundromat can be freely split/combined on each newline, since unescaped newlines do not occur in literals and blank nodes are replaced in a standards-compliant way by well-known IRIs.

Streaming: LOD Laundromat supports streamed processing of triples since it guarantees that no duplicate triples occur in the data stream. This means that the streamed processor does not have to perform additional bookkeeping in order to check for statements that were observed earlier.

Completeness: The LOD Laundromat guarantees that cleaned data is always a complete representation of the input dataset to the extent at which the original dataset is standards-compliant.

4 Use cases

The LOD Laundromat provides added value in a wide range of use cases. We here give four example:

1. From the scraping metadata statistics can be extracted that provide a (partial) overview of the state of the LOD Cloud (e.g., how much data is disseminated and how well this is done).
2. Since all errors are stored as part of the metadata, the LOD Laundromat gives detailed feedback to dataset publishers as to how they can improve the quality of their data.
3. By publishing very many datasets in exactly the same, standards-compliant way, LOD Laundromat supports the evaluation of SW algorithms on large-scale, heterogeneous and real-world data.
4. SW algorithms can use the scraping metadata in order to distribute data evenly over a given number of processing nodes (i.e., load balancing).

5 Conclusion

By using the LOD Laundromat we publish standards- and guideline-compliant datasets that are siblings of existing datasets. LOD Laundromat is a service which *continuously* crawls for additional datasets; the number of triples we publish already surpasses that of existing LOD collections such as the Billion Triple Challenge[5] and Freebase [1]. Because anybody can drop their dirty data in the LOD Laundry Basket the coverage of the LOD Laundromat will increase over time as more seedpoints are added.

References

[1] Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250. ACM, 2008.

[2] Aidan Hogan, Jürgen Umbrich, Andreas Harth, Richard Cyganiak, Axel Polleres, and Stefan Decker. An empirical survey of linked data conformance. *Web Semantics: Science, Services and Agents on the World Wide Web*, 14:14–44, 2012.

[5] http://km.aifb.kit.edu/projects/btc-2012/

Towards Aggression De-escalation Training with Virtual Agents: A Computational Model[1]

Tibor Bosse and Simon Provoost

VU University Amsterdam, Department of Computer Science
De Boelelaan 1081, 1081 HV Amsterdam, The Netherlands

1 Introduction

Aggressive behaviour against employees in the public sector, such as police officers, tram conductors, and ambulance personnel, is an ongoing concern worldwide. According to a recent study in the Netherlands, around 60% of the employees in the public sector has been confronted with undesired behaviour in the last 12 months [1]. Verbal aggression is experienced most frequently, by 57% of the respondents, followed by physical aggression (20%), intimidation (19%), discrimination (12%) and sexual intimidation (7%) in 2011.

Being confronted with (verbal) aggression has been closely associated with psychological distress, which in turn has a negative impact on work performance. Responses to aggression range from emotions like anger and humiliation through intent to leave the profession. Verbal aggression by customers may impair employees' recognition and working memory. In case of severe incidents, employees may even develop symptoms indicating post-traumatic stress syndrome.

2 Aggression De-escalation Training

To deal with aggression, a variety of techniques are available that may prevent escalation [2, 4]. These include (verbal and non-verbal) communication skills, conflict resolution strategies, and emotion regulation techniques. The current paper is part of a project that aims to develop a serious game for aggression de-escalation training, based on Virtual Reality. VR-based training has proven to be a cost-effective alternative for real world training in several domains, varying from surgery to negotiation.

In the training environment envisioned in the current project, a trainee will be placed in a virtual scenario in which aggression plays a role (e.g., handling a domestic violence case), with the goal to handle it as adequately as possible. She can observe the events that happen in the scenario (e.g., a virtual character starts offending her) via vision and sound, and has to respond to this by selecting the most appropriate action from a multiple choice menu. During the task, the trainee is being 'monitored' by an intelligent software system that observes her behaviour, analyses this, and provides personalised support.

---

[1] The full paper has been published as: Bosse, T. and Provoost, S. (2014), Towards Aggression De-escalation Training with Virtual Agents: A Computational Model. In: P. Zaphiris and A. Ioannou (eds.), Learning and Collaboration Technologies. Technology-Rich Environments for Learning and Collaboration, Part II. Lecture Notes in Computer Science, vol. 8524, Springer Verlag, pp. 375-387.

3 Computational Model

To realise an effective training system as described above, it is crucial to understand the dynamics of the processes related to interpersonal aggression. More specifically, when focussing on 1:1 interaction, knowledge is required about how aggression builds up in person A (the aggressor), and what person B (the de-escalator) can do to make it go down again. In the current paper, such knowledge is formalised in terms of a dynamical computational model of interpersonal aggression. Basically, this model consists of two separate sub-models, one for the aggressor and one for the de-escalator.

The *aggressor* model makes a distinction between reactive aggression (i.e., a response to a negative event that frustrates the person's goals) and proactive aggression (i.e., an instrumental type of aggression used to achieve a certain goal), cf. [3]. In addition, the dynamics of aggression are modelled as a cyclic process that passes through five separate phases (calm - threat perception - mounting anxiety - eruption of physical aggression - recovery), cf. [2]. The *de-escalator* model is based on a standard protocol used for training of employees in public services in the Netherlands [4]. This model prescribes appropriate reactions (e.g., letting go, calming down, showing understanding, increasing dominance, calling for support) for a variety of circumstances, which can be related to the five phases mentioned above.

To study the behaviour of the combined (aggressor-de-escalator) model, a number of simulation runs have been generated under different parameter settings. These results have pointed out that the model is capable of simulating different types of scenarios that correspond to cases that are observed in reality. Examples of such scenarios include cases where the de-escalator succeeds in calming down the aggressor (both for reactive and proactive types of aggressors), but also cases where the de-escalator fails to do so. In the latter case, there may be different reasons (e.g., the de-escalator fails to recognize the type and intensity of aggression of the conversation partner, or fails to control his own emotional state).

4 Conclusion

The results of this study are useful because the implemented models can be directly incorporated in the VR-based training system under development. In particular, the aggressor model will be used to control the behaviour of the 'aggressive virtual agents' that are displayed in the scenarios, whereas the de-escalator model will be used by the training system as a prescriptive model for adequate de-escalation. By comparing the behaviour of this de-escalator model with the actions performed by the trainee, the system will be able to make a detailed analysis of her performance, allowing it to provide personalised feedback in case of mistakes. Indeed, in follow-up research, both models will be integrated into our system and validated, thus providing a more theoretical foundation to VR-based training of aggression de-escalation.

References

[1] Abraham, M., Flight, S., and Roorda, W. (2011). Agressie en geweld tegen werknemers met een publieke taak. Onderzoek voor Veilige Publieke Taak 2007 - 2009 – 2011. Amsterdam: DSP.

[2] Anderson, L.N. and Clarke, J.T. (1996). De-escalating verbal aggression in primary care settings. Nurse Pract. 21(10):95, 98, 101-2.

[3] Dodge, K.A. (1990). The structure and function of reactive and proactive aggression. In: The development and treatment of childhood aggression (pp. 201-218). Hillsdale, NJ: Erlbaum.

[4] Ministry of the Interior and Kingdom Relations (2008). Handboek agressie en geweld - voorkomen, beperken, afhandelen. Technical Report for the Programme 'Veilige Publieke Taak'. April 2008.

Combining Multiple Correlated Reward and Shaping Signals by Measuring Confidence[1]

Tim Brys [a] Ann Nowé [a] Daniel Kudenko [b] Matthew E. Taylor [c]

[a] *Vrije Universiteit Brussel, Brussels, Belgium*
[b] *University of York, York, UK*
[c] *Washington State University, Pullman, WA*

Abstract

This paper falls in a line of research that investigates how reinforcement learning algorithms can be sped up by combining two AI techniques: multi-objectivization and ensemble techniques. Multi-objectivization is a process that turns a single-objective problem into a multi-objective one, essentially creating a diverse set of feedback signals for the original single-objective task. These signals can be strategically combined using ensemble techniques to speed up and improve learning. In this paper, we present a novel ensemble technique that detects where and when each of the signals is most informative, and proceeds to use that one for action selection.

1 Introduction

Multi-objective problems with correlated objectives are a class of problems that deserve specific attention. In contrast to typical multi-objective problems, they do not require the identification of trade-offs between the objectives, as (near-) optimal solutions for any objective are (near-) optimal for every objective. Intelligently combining the feedback from these objectives, instead of only looking at a single one, can improve optimization. This class of problems is very relevant in reinforcement learning, as any single-objective reinforcement learning problem can be framed as such a multi-objective problem using multiple reward shaping functions [2]. Previously, this kind of problem was solved using a linear scalarization of the signals, which can lead to many problems and impracticalities, including the need for weight tuning. In this paper, we present a parameterless, scale-invariant ensemble technique specifically built for this type of problems, called *adaptive objective selection*.

2 Adaptive Objective Selection

With adaptive objective selection [3], a learning agent estimates the Q-function for every objective o in parallel (thus learning $Q(s, a, o)$ values), and decides before every action selection decision which objective's estimates to use. To make this objective selection decision, we introduce the concept of *confidence* in learned estimates. We define confidence as an estimation of the likelihood that the estimates are correct. Higher-variance reward distributions will make any estimate of the average reward less confident, and always selecting the objective whose estimates are most likely to be correct will maximize the likelihood of correctly ranking the action set. This is related to the use of confidence intervals in UCB [1]. To measure confidence in estimates, the agent models actions as distributions, instead of just a mean (Q-value), and uses common statistical tests to estimate how confident he can be about the estimates of each objective

Adaptive objective selection has several interesting properties. It makes its decisions a function of the state-space, which can account for different objectives being more or less reliable in different parts of

[1] "The full paper has been published in *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, 2014."

the state space. Furthermore, it uses the objectives in a scale-invariant way. That is, its workings do not depend on the relative scalings of the objectives, since all statistical tests proposed are scale-invariant, and thus no parameters are introduced.

The most interesting variant of adaptive objective selection proposed exploits tile-coding function approximation. Tile-coding provides a very natural way to represent actions as distributions, without requiring the storage of extra information. For a triplet (s, a, o), the agent can simply take the weights in $\theta_{o,a}$ activated by s as samples representing the distribution of that triplet. Then we can estimate confidence by applying a paired statistical test to the samples of every action (or of the estimated best and worst actions). We can use a paired test, such as the paired Student's t-test or the Wilcoxon signed-rank test, because the weights for different actions and objectives will come from the same tiles, i.e. locations, in the same tilings, although stored in different weight vectors.

3 Experiments

We experimentally validated the technique on two problems: a traffic light control problem [4] and the pursuit domain [6]. In both domains, using adaptive objective selection yields significant improvements in performance (final as well as cumulative), as evidenced by faster convergence to better policies [3]. Furthermore, the objective selection decisions are shown to make intuitive sense, and demonstrate that confidence increases as the heuristic knowledge aligns with the original objective, while the reverse happens as the heuristic knowledge is contradicting the actual objective, leading the learning agent to exploit the best heuristic knowledge in any situation.

In more recent work, we have shown that more classical ensemble techniques [7] for reinforcement learning also work well on multi-objectivized problems [5], adding credence to the combination of multi-objectivization and ensemble techniques being a viable approach to solve standard reinforcement learning problems faster and better.

4 Acknowledgements

Tim Brys is funded by a Ph.D grant of the Research Foundation-Flanders (FWO). This work was supported in part by NSF IIS-1149917 and NSF IIS-1319412.

References

[1] Peter Auer, Nicolò Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2-3):235–256, 2002.

[2] Tim Brys, Anna Harutyunyan, Peter Vrancx, Matthew E. Taylor, Daniel Kudenko, and Ann Nowé. Multi-objectivization of reinforcement learning problems by reward shaping. In *Proceedings of the International Joint Conference on Neural Networks*. IEEE, 2014.

[3] Tim Brys, Ann Nowé, Daniel Kudenko, and Matthew E. Taylor. Combining multiple correlated reward and shaping signals by measuring confidence. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence (AAAI-14)*, 2014.

[4] Tim Brys, Tong T. Pham, and Matthew E. Taylor. Distributed learning and multi-objectivity in traffic light control. *Connection Science*, 26(1):56–83, 2014.

[5] Tim Brys, Matthew E. Taylor, and Ann Nowé. Using ensemble techniques and multi-objectivization to solve reinforcement learning problems. In *Proceedings of the European Conference on Artificial Intelligence (ECAI-14)*, 2014.

[6] Peter Stone and Manuela Veloso. Multiagent systems: A survey from a machine learning perspective. *Autonomous Robots*, 8(3):345–383, 2000.

[7] Marco A Wiering and Hado van Hasselt. Ensemble algorithms in reinforcement learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 38(4):930–936, 2008.

Decomposition of Intervals in the Space of Anti-Monotonic Functions *

Patrick De Causmaecker[1] and Stefan De Wannemacker[1,†]

[1]CODeS & iMinds-ITEC, Department of Computerscience, KULAK, KU Leuven
Patrick.DeCausmaecker@kuleuven-kulak.be
[†]In honor of Stefan who passed away December 2013

Keywords
Boolean functions, complete distributive lattices, Dedekind numbers, anti-chains, Sperner families

1 Motivation

The space of (anti-)monotone Boolean functions has many applications [2]. Below we give some examples. The motivation for our study was to design a topology to describe the behavior of such functions. This lead us to to intervals in the associated lattice. In order to understand their structure, we studied a well known counting problem. Our counting procedures required several new insights in the structure of intervals. We discover new recursion relations, partitioning the space and efficiently counting the number of elements. This work was published as [1].

2 Examples

2.1 Sets of applications that make a computer crash

Given a set of n applications that can be run on one computer. The observation is that certain combinations of applications make a computer crash. A reasonable assumption is that if the computer crashes for a set S of applications, the computer will crash for each superset of S, and, if a computer does not crash on S, it will not crash on any subset of S. The function that decides whether a computer crashes on a set or not is hence a monotone boolean function. Predicting the behavior of a computer on the set of applications can rely on learning this function [5].

2.2 Failing students

Determining whether a student fails or not depends on the set of courses for which the student passes or not. Any jury decision should be consistent. If one student fails with a given set S of non-passes, no student with a superset of S of non-passes should pass, and if a student passes with a given set S of non-passes, every student with a subset of S of non-passes should pass. The judgement can be modeled by monotonic Boolean functions.

2.3 Hamiltonian cycle

The Hamiltonian cycle problem asks for the existence of a graph cycle through a graph visiting each node exactly once. If a graph allows a Hamiltonian cycle, each super graph allows a Hamiltonian cycle. The decision function deciding this problem is a monotone Boolean function.

*Work supported by the Belgian Science Policy Office (BELSPO) in the Interuniversity Attraction Pole COMEX. (http://comex.ulb.ac.be)

3 Aim

The motivation for this research is to find fundamental objects for the study of monotone Boolean functions. We believe intervals in the corresponding lattices are such objects. They could be at the basis of a topology. In this paper we study their structure as part of a counting exercise [4].

Monotone Boolean functions are notoriously hard to manipulate in this context. This is why we introduced the concept of anti-monotonic functions. These correspond to the minsets of monotone functions; they take the value true on a set S only if the monotone function takes the value true on S and false on any real subset of S. The sets accepted by an anti-monotone function form an antichain. The study of monotone Boolean functions on an n-set is equivalent to the study of anti-monotone functions and to the study of anti-chains in an n-set.

4 Results

For two anti-monotone functions we say $\alpha \leq \beta \Leftrightarrow$ for each set A such that $\alpha(A) = true$, there is a superset B for which $\beta(B) = true$. Intervals are defined as $[\alpha, \beta] = \{\chi | \alpha \leq \chi \leq \beta\}$ and analogously for open and half-open intervals.

Associate a graph with each interval. Its vertices are the sets in the top of the interval. Two vertices are connected by an edge in the graph if and only if the intersection of the two sets is not dominated by a set in the bottom of the interval.

Result 1. *The connected components of the associated graph of an interval determine its decomposability as a direct join.*

For a number l, level l in an interval is the anti-monotonic function accepting all sets of size l.

Result 2. *Levels allow a power of two expansion for the size of an interval.*

Considering specific homomorphisms between the full lattice of monotone functions on an n-set and the k-dimensional hypercube leads to

Result 3.

$$Dedekind\ number\ of\ order\ n+k = \sum_{\rho_0 \geq \rho_N, fucntions\ on\ N} |[\{\}, \rho_N]| P_{n,k,\rho_0,\rho_N} |[\rho_0, \{N\}]|$$

The P-coefficients are hard to compute in general. In case $k = 2$, $P_{n,k,\rho_0,\rho_N} = 2^{C_{[\rho_N,\rho_0]}}$ where $C_{[\rho_N,\rho_0]}$ denotes the number of connected components in the graph associated with the interval.

References

[1] De Causmaecker, P., De Wannemacker, S. Decomposition of Intervals in the Space of Anti-Monotonic Functions. In Ojeda-Aciego, M. (Ed.), Outrata, J. (Ed.), Proceedings of the Tenth International Conference on Concept Lattices and Their Applications (CLA 2013), La Rochelle, France, October 15-18, 2013.: Vol. 1062. (pp. 57-67).

[2] Y. Crama and P.L. Hammer, Boolean functions: Theory, algorithms, and applications, Encyclopedia of Mathematics and Its Applications, Cambridge University Press, 2011.

[3] Sloane, N. J. A., The On-Line Encyclopedia of Integer Sequences. (OEIS), *http://www.research.att.com/ njas/sequences/.*

[4] Dedekind, Richard (1897), Über Zerlegungen von Zahlen durch ihre größten gemeinsamen Teiler, Gesammelte Werke, 2, pp. 103-148.

[5] S. Kurz: Competitive learning of monotone Boolean functions (extended abstract), in Suhl, Leena; Mitra, Gautam et al. (Hrsg.): Applied Mathematical Optimization and Modelling, APMOD 2012 Extended Abstracts, DS&OR Lab, Band 8, University of Paderborn, Germany 2012, Seiten 416-421

Probabilistic Argumentation Frameworks – A Logical Approach[1]

Dragan Doder [a] Stefan Woltran [b]

[a] *Computer Science and Communications, University of Luxembourg*
Rue Richard Coudenhove-Kalergi 6, L-1359 Luxembourg
[b] *Institute of Information Systems, Vienna University of Technology*
Favoritenstrasse 9–11, 1040 Vienna, Austria

Within the last decade, abstract argumentation has emerged as a central field in Artificial Intelligence. Hereby, it is only the relation between arguments which is taken into account when evaluating a certain scenario; the actual contents of the arguments do not play a role. The most simple objects used in abstract argumentation are Dung's argumentation frameworks (AFs). AFs are just directed graphs where vertices represent the arguments and edges indicate a certain conflict between the two connected arguments. The goal is to identify jointly acceptable sets of arguments for which a large selection of different semantics is available. One particular line of research in abstract argumentation concerns the formalization of such argumentation semantics in terms of logics, see e.g. [1, 6, 7]. Such formalizations not only provide a uniform definition of the different semantics, they also can lay the foundations for efficient systems (see e.g. [2, 4] where SAT-solvers are used to evaluate formulas of propositional logic which express certain argumentation problems). Another benefit of logically characterizing argumentation problems is the potential of direct derivation of important properties (for example, complexity results based on characterizations in Monadic Second-Order Logic are given in [3, 5]).

For many applications, Dung's AFs appear too simple in order to conveniently model all aspects of an argumentation problem. One such shortcoming is the lack of handling levels of uncertainty, an aspect which typically occurs in domains, where diverging opinions are raised. This calls for augmenting simple AFs with probabilities. Several proposals have been recently made. The most detailed overview is probably the article by Hunter [8].

As for standard abstract argumentation, a uniform logical formalization for AFs with probabilities is of great help in order to understand and compare these different approaches. Not surprisingly, the most suitable logic for this purpose is probabilistic logic [10]. In this paper, we want to take a first step towards such a unified view. Actually, we focus here on the approach of Li, Oren and Norman [9]. Hereby, an argumentation framework is enriched by probabilities assigned to both arguments and conflicts. These probabilities are then used to calculate probabilities of subgraphs of the given AF via the "independency" assumption. The probability for a given set S to be an extension (with respect to a particular semantics) is obtained from the sum of the probabilities of the subgraphs for which S is such an extension in the standard way. Our main goal is to express this entire process by logical means.

In this work we extended ideas of Besnard and Doutre [1], who addressed the problem of acceptability of sets of arguments under Dung's semantics via propositional logic, to probabilistic argumentation. We have focused on the approaches from [9, 8], where argumentation frameworks $\langle A, R\rangle$ are enriched with probabilities on A and R. Those probabilities determine certain probabilities of subgraphs of $\langle A, R\rangle$, using the "independency" assumption. We have provided a logical formalization in terms of propositional probabilistic logic using set of formulae depending on a given PAF and class of models that correspond to the probability measures induced by PAFs.

More specifically, our main contributions are as follows:

[1]The full paper is published in *Proceedings of the The 8th International Conference on Scalable Uncertainty Management (SUM 2014)*, volume 8720 of LNCS, pages 134–147. Springer, 2014.

- We start with formulae which allow us to handle subgraphs of AFs in a direct way. Given a semantics σ, an AF $F\langle A, R\rangle$ and a set $\Gamma \subseteq A$, we provide the formula $\psi_F^\sigma(\Gamma)$ of the appropriate propositional language, whose models correspond to the elements of set $Q_F^\sigma(\Gamma)$ of all subgraphs of F for which Γ is a σ extension. These encodings will be the basis for our characterizations in terms of probabilistic logic. Note that in contrast to the encodings from [1] (where the framework remains fixed and the extensions are characterized by models), we fix here the extension and characterize the appropriate subgraphs.
- We build a probabilistic logic suitable for modelling probabilistic argumentation frameworks. In order to define semantics for the logic, we characterize the class of probability measures induced by the "independence" assumption. We apply the probabilistic operators to the constructed propositional formulas in order to formally represent the probabilities of different extensions.
- Given an AF $F = \langle A, R\rangle$, we define the natural correspondence between PAFs which are obtained by assigning probabilities to A and R, and models of probabilistic logic. Then, we associate to $\Gamma \subseteq A$, σ and $r \in [0, 1]$ the probabilistic formula $\rho_F^\sigma(\Gamma, r)$ such that the models of $\rho_F^\sigma(\Gamma, r)$ correspond to probabilistic argumentation frameworks for which Γ is σ extension with probability at least r. We also check if a set of arguments is acceptable with a given probability (w.r.t. a given semantics) by checking satisfiability of the corresponding formula.

Although we follow the initial paper on probabilistic argumentation frameworks [9], it is easy to modify our results to cover the more general case, where the independency assumptions are not used.

Acknowledgment

This work was supported by the National Research Fund (FNR) of Luxembourg through project PRIMAT, and by the Austrian Science Fund (FWF): project I1102.

References

[1] Philippe Besnard and Sylvie Doutre. Checking the acceptability of a set of arguments. In *Proc. NMR 2004*, pages 59–64, 2004.

[2] Frederico Cerutti, Paul E. Dunne, Massimiliano Giacomin, and Mauro Vallati. A SAT-based Approach for Computing Extensions on Abstract Argumentation. In *Proc. TAFA*, 2013.

[3] Paul E. Dunne. Computational properties of argument systems satisfying graph-theoretic constraints. *Artif. Intell.*, 171(10-15):701–729, 2007.

[4] Wolfgang Dvořák, Matti Järvisalo, Johannes Peter Wallner, and Stefan Woltran. Complexity-sensitive decision procedures for abstract argumentation. *Artif. Intell.*, 206:53–78, 2014.

[5] Wolfgang Dvořák, Stefan Szeider, and Stefan Woltran. Abstract argumentation via monadic second order logic. In *Proc. SUM*, volume 7520 of *LNCS*, pages 85–98. Springer, 2012.

[6] Uwe Egly and Stefan Woltran. Reasoning in argumentation frameworks using quantified boolean formulas. In *Proc. COMMA*, volume 144 of *FAIA*, pages 133–144. IOS Press, 2006.

[7] Davide Grossi. Argumentation in the view of modal logic. In *Proc. ArgMAS*, volume 6614 of *LNCS*, pages 190–208. Springer, 2010.

[8] Anthony Hunter. A probabilistic approach to modelling uncertain logical arguments. *Int. J. Approx. Reasoning*, 54(1):47–81, 2013.

[9] Hengfei Li, Nir Oren, and Timothy J. Norman. Probabilistic argumentation frameworks. In Sanjay Modgil, Nir Oren, and Francesca Toni, editors, *Proc. TAFA*, volume 7132 of *LNCS*, pages 1–16. Springer, 2012.

[10] Zoran Ognjanović and Miodrag Rašković. Some probability logics with new types of probability operators. *J. Log. Comput.*, 9(2):181–195, 1999.

Structural Properties as Proxy for Semantic Relevance in RDF Graph Sampling[1]

Laurens Rietveld[a] Rinke Hoekstra[ab] Stefan Schlobach[a]
Christophe Guéret[c] Wouter Beek[a]

[a] Dept. of Computer Science, VU University Amsterdam, NL {laurens.rietveld,k.s.schlobach,rinke.hoekstra,w.g.j.beek}@vu.nl [b] Leibniz Center for Law, University of Amsterdam, NL, hoekstra@uva.nl [c]Data Archiving and Network Services (DANS), NL, christophe.gueret@dans.knaw.nl

Abstract

The Linked Data cloud has grown to become the largest knowledge base ever constructed. Its size is now turning into a major bottleneck for many applications. In order to facilitate access to this structured information, this paper proposes an automatic sampling method targeted at maximizing answer coverage for applications using SPARQL querying. The approach presented in this paper is novel: no similar RDF sampling approach exist. Additionally, the concept of creating a sample aimed at maximizing SPARQL answer coverage, is unique. We empirically show that the relevance of triples for sampling (a semantic notion) is influenced by the topology of the graph (purely structural), and can be determined without prior knowledge of the queries. Experiments show a significantly higher recall of topology based sampling methods over random and naive baseline approaches (e.g. up to 90% for Open-BioMed at a sample size of 6%).

1 Introduction

The Linked Data cloud grows every year and has turned the Web of Data into a knowledge base of unprecedented size and complexity. This poses problems with respect to the scalability of our current infrastructure and tools. Datasets such as DBPedia (459M triples) and Linked Geo Data (289M triples) are central to many Linked Data applications. Local use of such large datasets requires investments in powerful hardware, and cloud-based hosting is not free either. These costs are avoidable if we know which part of the dataset is needed for our application, i.e. if only we could pick the data a priori that is actually being used, or required to solve a particular task. Under several circumstances, e.g. for prototyping, demoing or testing, developers and users are content with relevant subsets of the data, and accept the possibility of incomplete results that comes with it. A locally available subset is also useful when the connection to a cloud based server is inaccessible (something which happens frequently [1]).

Our analysis of five large datasets (>50M triples) shows that for a realistic set of queries, at most 2% of the dataset is actually used: a clear opportunity for pruning RDF datasets to more manageable sizes. Unfortunately, this set of queries is not always known: queries are not logged or logs are not available because of privacy or property rights issues. And even if a query set is available, it may not be representative or suitable, e.g. it contains queries that return the entire dataset.

We define *relevant sampling* as the task of finding those parts of an RDF graph that maximize a task-specific relevance function while minimizing size. For our use case, this relevance function relies on semantics: we try to find the smallest part of the data that entails as many of the original answers to

[1]This work was supported by the Dutch national program COMMIT, and carried out on the Dutch national e-infrastructure with the support of SURF Foundation. The full paper has been published in *Proceedings of International Semantic Web Conference*, 2014.

typical SPARQL queries as possible. This paper investigates whether we can use structural properties of RDF graphs to predict the relevance of triples for typical queries.

To evaluate this approach, we represent "typical use" by means of a large number of SPARQL queries fired against datasets of various size and domain: *DBpedia 3.9*, *Linked Geo Data*, *MetaLex*, *Open-BioMed*[2], *Bio2RDF* and *Semantic Web Dog Food*. The queries were obtained from server logs of the triple stores hosting the datasets and range between 800 and 5000 queries for each dataset. Given these datasets and query logs, we then 1) rewrite RDF graphs into directed unlabeled graphs following 5 different rewrite strategies, 2) analyze the topology of these graphs using three standard network analysis methods (In Degree, Out Degree and PageRank), 3) assign the derived weights to triples, and 4) generate samples for every percentile of the size of the original graph. These steps were implemented as a scalable sampling pipeline, called *SampLD*.[3] SampLD allows us to evaluate 15.600 different combinations of datasets, rewritten graphs, network analysis and sample sizes.

Our results show that the topology of the hypergraph alone helps to predict the relevance of triples for typical use in SPARQL queries. In other words, we show in this paper that without prior knowledge of the queries to be answered, we can determine to a surprisingly high degree which triples in the dataset can safely be ignored and which cannot. As a result, we are able to achieve a recall of up to .96 with a sample size as small as 6%, using *only* the structural properties of the graph. This means that we can use purely *structural* properties of a knowledge base as proxy for a *semantic* notion of relevance.

Because the datasets we use are quite large, ranging up to 459 Million triples for DBPedia, each of these steps was implemented using libraries for scalable distributed computing (Pig and Giraph). Scale also means that we are restricted to network metrics that are parallelizable.

2 Conclusion

RDF graph topology, query type and structure, sample size; each of these can influence the quality of samples produced by a combination of graph rewriting and network analysis. For sample sizes 10%, 25% and 50%, the majority of the best performing sampling methods have significantly better average recall ($\alpha = 0.05$) than the random sample. Exceptions are LGD and Bio2RDF for sample size 10%, and MetaLex for sample size 10% and 25%. In DBPedia, a sample size of 7% based on a path-based rewrite and PageRank sampling method already results in an average recall of 0.5. An interactive overview of our results, including recall plots, significance tests, and degree distributions for *every* dataset, is available online.[4]

This paper does not offer a definitive answer as to which combination is the best fit: we cannot yet predict the best performing sampling method given a data and query set. To make this possible, we plan to use Machine Learning on the results for several more independent data and query sets. Although SampLD provides the technical means, the number of publicly available query sets is currently too limited to learn significant correlations (6 query sets in total for USEWOD 2013, only 3 in 2014)[5].

Our results indicate that the topology of RDF graphs *can* be used to determine good samples that, in many cases, significantly outperform our baselines. Indeed, this shows that we can mimic *semantic* relevance through *structural* properties of RDF graphs, without an a-priori notion of relevance.

References

[1] Carlos Buil-Aranda, Aidan Hogan, Jürgen Umbrich, and Pierre-Yves Vandenbussche. Sparql web-querying infrastructure: Ready for action? In *The Semantic Web–ISWC 2013*, pages 277–293. Springer, 2013.

[2] See `http://www.open-biomed.org.uk/`

[3] The SampLD pipeline and evaluation procedure are available online at `https://github.com/Data2Semantics/GraphSampling/`

[4] See `http://data2semantics.github.io/GraphSampling/`

[5] See `http://data.semanticweb.org/usewod/2013/challenge.html` and `http://usewod.org/reader/release-of-the-2014-usewod-log-data-set.html`, respectively

Feasibility Estimation for Clinical Trials [1]

Zhisheng Huang [a] Frank van Harmelen [a] Annette ten Teije [a]

Andre Dekker [b]

[a] *VU University Amsterdam, The Netherlands*,
{huang,Frank.van.Harmelen,annette}@cs.vu.nl
[b] *MAASTRO Clinic, The Netherlands*, andre.dekker@maastro.nl

Abstract

At least 90% of trials are extended by at least 6 weeks because investigators fail to enroll patients on schedule. It is therefore important at trial design-time to have good insight in how the choice of the eligibility criteria affects the recruitment rate. Based on that insight, trial designers can then adjust the eligibility criteria in order to ensure realistic recruiting rates. In this paper we propose a simple mathematical model to determine how eligibility criteria determine the recruitment rate. We have implemented this mathematical model in efficient algorithms, and we demonstrate our model on both real and synthetic patient data. Our experiments show that almost all medical trials in our test corpus contain superfluous criteria, and that this redundancy can only be revealed with our new relative feasibility measure (and not with the classical absolute feasibility measure). To increase the reproducibility of our results, we have made our datasets available online.

1 Introduction

Motivation. Trial recruitment is challenging for medical researchers, who frequently overestimate the pool of qualified, willing participants. If the target sample size is not achieved, the trial has less statistical power to detect potentially important clinical differences between the groups, so the results may be less useful. In addition, if recruitment has to be extended to reach the required sample size, the trial will cost more and take longer, delaying the use of the results in clinical practice.

Research Question. Given these persistent and costly problems, it is clearly important that trial designers are given the tools to perform an accurate estimate of *trial feasibility* including accurate estimates of recruiting rates. Such tools will help them to avoid overly restrictive trial criteria, thereby avoiding low recruitment rates. This leads us to the central question of this paper: *given the characteristics of the patient population, which trial conditions will lead to which cohort size?*

Approach. In this paper, we will distinguish two kinds of trial feasibility: absolute feasibility and relative feasibility. The former considers the effect of a total set of conditions, whereas the latter considers the effect of adding, removing or changing individual conditions in the presence of other conditions. We propose a simple mathematical model of trial feasibility to explore the distinction between absolute feasibility and relative feasibility. This will show that our novel notion of relative feasibility is a useful notion in the design of a clinical trial.

A workflow of trial feasibility usually consists of the following steps: (1) start with new or existing trial design, (2) determine required cohort size (statistical power), (3) determine absolute feasibility of current design, (4) explore relative impact of modifying some conditions, (5) repeat steps 3-4.

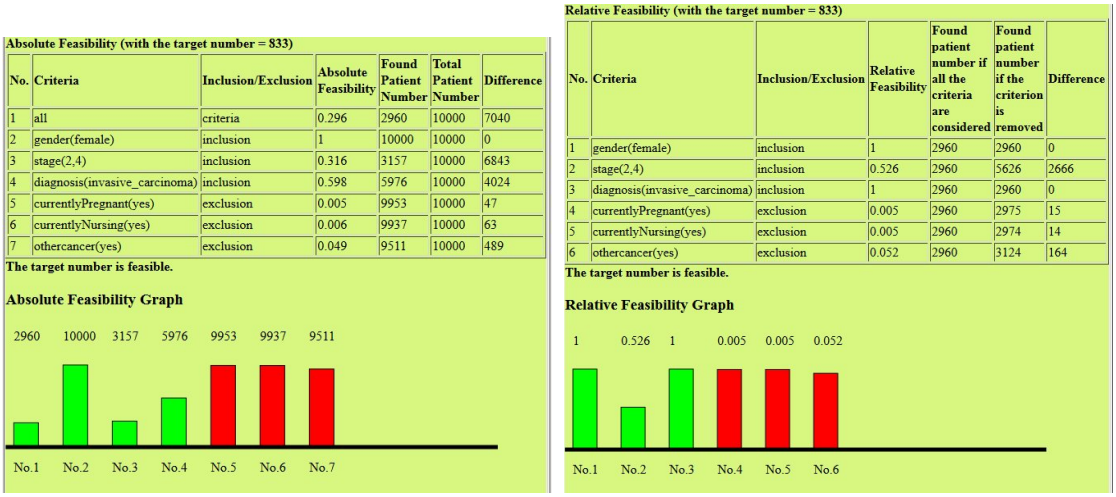
This workflow has been implemented and integrated with SemanticCT, a semantically-enabled system for clinical trials. We have conducted several experiments to test our approach to trial feasibility analysis with a set of real patient data at a clinic in the Netherlands, and with a set of synthetic patient data, which are generated by using a knowledge-based patient data generator.

[1] The full paper has been published in *Proceedings of the 7th International Conference on Health Informatics (HEALTHINF2014)*, pages 68–77, France, 3-6 march, 2014.(Best Paper Award).

Using these data, we performed 16 experiments of calculating trial feasibility with different criteria, different parameters in those criteria, and different cohort sizes.

For example, in the experiment no.1, we select the trial NCT00001385 as the template, with a cohort size of 200, a consent rate of 30% and a dropout rate of 20%, leading to the target number 833.

The experiment result shows that the target number is feasible with the absolute feasibility values shown in Figure 1(a). Actually the system finds 2960 eligible patients for those criteria from the 10,000 patients in dataset ZSH2013A. Notice that 'gender(female)' has an absolute feasibility of 1, which means that all patients satisfy this condition. In the absolute feasibility graph shown in Figure 1(a), a high score on an inclusion criterion means that the criterion alone is not very selective, in other words most patients are included. Similarly, a high score for an exclusion criterion means that only very few patients are excluded only based on that particular criterion, for instance only 47 patients are excluded by only taking into account the pregnancy condition.



Absolute Feasibility (with the target number = 833)

No.	Criteria	Inclusion/Exclusion	Absolute Feasibility	Found Patient Number	Total Patient Number	Difference
1	all	criteria	0.296	2960	10000	7040
2	gender(female)	inclusion	1	10000	10000	0
3	stage(2,4)	inclusion	0.316	3157	10000	6843
4	diagnosis(invasive_carcinoma)	inclusion	0.598	5976	10000	4024
5	currentlyPregnant(yes)	exclusion	0.005	9953	10000	47
6	currentlyNursing(yes)	exclusion	0.006	9937	10000	63
7	othercancer(yes)	exclusion	0.049	9511	10000	489

The target number is feasible.

Absolute Feasibility Graph

Relative Feasibility (with the target number = 833)

No.	Criteria	Inclusion/Exclusion	Relative Feasibility	Found patient number if all the criteria are considered	Found patient number if the criterion is removed	Difference
1	gender(female)	inclusion	1	2960	2960	0
2	stage(2,4)	inclusion	0.526	2960	5626	2666
3	diagnosis(invasive_carcinoma)	inclusion	1	2960	2960	0
4	currentlyPregnant(yes)	exclusion	0.005	2960	2975	15
5	currentlyNursing(yes)	exclusion	0.005	2960	2974	14
6	othercancer(yes)	exclusion	0.052	2960	3124	164

The target number is feasible.

Relative Feasibility Graph

(a) Absolute (b) Relative

Figure 1: Absolute and relative feasibility of experiment no.1

Our experiments reveal a number of interesting findings. When measured over realistic patient data, every one of the realistic trials that we looked at contains criteria that are strictly speaking superfluous, because when removed from the trial definition, the same set of realistic patients would have been recruited anyway. For exclusion criteria, essentially the same was observed. Secondly relative feasibility is more revealing than absolute feasibility. Thirdly inclusion criteria are more selective than exclusion criteria.

2 Conclusion

In this paper, we have developed a lightweight model of trial feasibility. Our model distinguishes the traditional notion of absolute feasibility from a new notion of relative feasibility. Absolute feasibility simply determines the number of eligible patients of a criterion. Relative on the other hand computes how the removal of a single criterion from a set of criteria affects the recruiting rate of the remaining criteria. We have implemented our lightweight mathematical model as part of the SemanticCT system, and have used it to determine the relative feasibility of different criteria for a number of different real-life trials, on both actual and synthetic (but realistic) patient data. Every trial we looked at contains criteria that could have been safely removed from the trial without loss of specificity. Furthermore, it seems that inclusion criteria typically contibute much more to the specificity of a trial then exclusion criteria. Those results could not have been found by using only the classical absolute feasibility measure. They can only be revealed by using our new relative feasibility estimator.

Acknowledgements This work is partially supported by the European Commission under the 7th framework programme EURECA Project (FP7-ICT-2011-7, Grant 288048), and by euroCAT (IVA Interreg).

Virtual Reflexes [1]

Catholijn M. Jonker[a] Joost Broekens[a] Aske Plaat[b]

[a] *Delft University of Technology, Mekelweg 4, Delft*
[b] *Leiden University, Niels Bohrweg 1, Leiden*

Abstract

Virtual Reality is used successfully to treat people for regular phobias. A new challenge is to develop Virtual Reality Exposure Training for social skills. Virtual actors in such systems have to show appropriate social behavior including emotions, gaze, and keeping distance. The behavior must be realistic and real-time. Current approaches are usually based on heavy information processing in terms of behavior planning, scripting of behavior and use of predefined animations. We believe this limits the directness of human bodily reflexes and causes unrealistic responses and delay. We propose to investigate virtual reflexes as concurrent sensory-motor processes to control individual parts of the virtual actor's skeleton with a body integrity model that keeps the effects coherent. We explain how emotion and cognitive modulation could be embedded, and give an example description of the interplay between emotion and a reflex.

1 Introduction

The idea of using virtual reality to treat regular phobia—such as fear of heights, or flying—relies on the success with which we can immerse the patient in a virtual situation. Patients should have an experience corresponding to a real life situation. For training social skills, the VR setting has to be such that the emotions and intentions that trainees would attribute to a real person are now attributed to a Virtual Character (VC).

To produce appropriate social virtual character behavior, typically heavy processing is involved in deciding based on sensory input generated by the trainee what responses need to be generated at the skeleton level of the VC. In particular, pattern recognition, reasoning and behavior planning, and virtual character animations need to be addressed. In contrast to this processing intensive approach, in this paper we propose a low-level approach based on the idea of virtual reflexes, in which observations directly cause VC muscle actuations with limited intermediate processing. Based on various primitive inputs, the muscle actuations are immediate and create bodily reactions. This generates fluent and fast responses in the VC. Perceived emotions emerge out of the interaction between sensory and motor information. Although no cognitive intermediate processing takes place, cognition and affect do modulate the sensory motor loops. As happens in the human body, various virtual reflexes can occur simultaneously and operate as concurrent subsystems. The contributions of the paper are as follows:

- We introduce an architecture for virtual reflexes.
- We link the architecture to neuropsychological theories on emotion & cognition.
- We formalize part of the reflexes in a virtual aggression training case study.

2 Virtual Reflex Architecture

The basis of our architecture is the uncoupling of various sensor-actuator channels. Behavior is generated by reflex nodes that dynamically couple sensory input and motor output. To cope with high-level influences on behavior, such as training scenarios, activity of these nodes is modulated by cognitive and

[1] In The full paper has been published in *Proceedings of the 14th International Conference on Intelligent Virtual Agents 2014*.

emotional factors. Each reflex node can be seen as a small control node that influences body parts. Its output is based on whether its sensory input deviates from a preset baseline, much like drives would need to be met in a homeostatic approach [1]. Upon deviation, three things happen concurrently. First, the deviation triggers activation of the body parts coupled to the reflex node. Second, the deviation has an effect on the emotional state. We envision a Pleasure–Arousal–Dominance (PAD) representation [2] of the emotional state (Emotion, in figure 1). Third, the deviation is available for cognition to reason upon.
The purpose of our work is to achieve immersive and realistic virtual environments for social skill training. To realize this, we propose a computational model for virtual character behavior based on parallel virtual reflexes that involve limited processing and operate directly on parts of the VC skeleton based on sensory input. Emotions emerge out of this interaction between sensory and motor information. Cognition is envisioned to modulate the sensory-motor loop. Virtual reflexes can occur simultaneously. Current limitations (we consider to be open questions) to our approach are (a) the ability of the body integrity model to enforce coherence, (b) the ability to generate scenario-relevant modulations of behavior solely based on cognitive modulation of parallel sensory motor loops, and (c) uncertainty about the emotion dynamics induced by a bidirectional relation between emotion and sensory-motor activation.

References

[1] Cañamero, L. 2005. Designing Emotional Artifacts for Social Interaction: Challenges and Perspectives. In L. Cañamero, R. Aylett (Eds.), *Animating Expressive Characters for Social Interaction*. Adv in Consciousness Research.

[2] Mehrabian, A. (1980). Basic Dimensions for a General Psychological Theory: OG&H.

Autonomous E-Coaching in the Wild: Empirical Validation of a Model-Based Reasoning System[1]

Bart A. Kamphorst [a] Michel C.A. Klein [b] Arlette van Wissen [b]

[a] *Utrecht University, Janskerkhof 13-13A, 3512 BL Utrecht*
[b] *VU University Amsterdam, De Boelelaan 1085, 1081 HV, Amsterdam*

1 Introduction

Advances in pervasive computing and agent systems are opening up new possibilities for intelligent decision support. Considering their potential impact on people's decision making and behavior, validation of these systems — especially when they are concerned with sensitive domains such as health or safety — is essential and deserves attention. This work aims to evaluate and validate an intelligent support system for behavior change. eMate is a versatile e-coaching system designed to coach people towards lasting behavior change in health domains such as maintaining a healthy diet, regulating medicine intake and increasing exercise [2]. It relies on a model of behavior change, called the COMBI model, which consists of constructs (including, but not limited to, motivation, barriers, coping, awareness, social norms and commitment) that represent the cognitive and emotional states of a user related to different stages of behavior change (which are consecutively *precontemplation* (PC), *contemplation* (C), *preparation* (P), *action* (A), and *maintenance* (M)). Note that only the latter two stages represent actual changes in behavior. eMate uses rule-based reasoning with the COMBI model to hypothesize about the *stage of change* of a coachee and the constructs that cause non-adherence.

2 Experimental Setup

A monthlong experimental study was performed, in which 82 students received support to take the stairs more often. The performance of the system is evaluated with regard to a) the method of determining the content of the support and b) the effect of the support. All participants were asked to monitor the number of stairs they climbed using the eMate app together with an Android widget for four weeks. After one week, a new goal (a 10% increase) was set, based on individual behavior in that first week. Participants then received three weeks of remote coaching to reach that goal. They received questions and motivational messages on the eMate smartphone app and on the eMate website. On that same website, participants could monitor their progress and see a motivational picture relating to the number of stairs they had already climbed.

Participants were administered an intake as well as a post questionnaire. Both questionnaires were filled out online and consisted of several validated surveys as well as some single items, all pertaining to the model constructs.[2] The answers from the intake questionnaire served as initial data points for the model. Answers pertaining to the same construct were aggregated and scaled to a value between 0–10 (where a higher score is better). Upon completing the post questionnaire, participants were payed €10.

[1] The full paper has been published in *Proceedings of the 13th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2014)*, pages 725–732, 2014 [1].

[2] All used surveys (in Dutch) can be found at `http://bit.ly/stairs_surveys`.

3 Results

Hypothesis 1: eMate identifies and accurately targets the problematic constructs for an individual. Problematic constructs are defined as constructs with low values that are preventing behavior change. Identifying and accurately targeting a person's problematic constructs is an important aspect of eMate's coaching strategy. To evaluate whether eMate's model-based diagnosis and targeting was accurate, the list of problematic constructs was (re)generated on the basis of 1) the intake questionnaire and 2) the results of eMate's model-based reasoning process. This list was compared to the targets of the messages that were sent to the coachees. We expected that each targeted construct was an element of the set of problematic constructs. Indeed, we found that this was the case for all messages that were received. Moreover, the mean value of constructs that were targeted were significantly lower compared to the mean of constructs that were not targeted (4.70 and 7.59 respectively, t-test: $p<0.001$).

Hypothesis 2: the total intervention (i.e. monitoring and coaching) has no clear negative effects. For successful e-coaching over longer periods of time, it is important that the coaching does not have unexpected negative effects. Results show that three constructs were improved significantly by the coaching: barriers, coping and motivation. Contrary to $H2$, there was a significant decrease of the value for the construct of social norms. However, we suspect that this difference is caused by initial overly optimistic answers by the coachees, but see [1] for a full discussion. Most coachees did not change with regard to their levels of awareness and commitment. This is explained by the fact that all but one coachee gave the highest possible ranking for awareness and for commitment at the start of the experiment.

Hypothesis 3: targeting by eMate improves construct values and promotes stage progression. We expected the targeted interventions to positively affect the problematic constructs and people's progress through stages. No significant differences were found between the stage in which coachees began and ended (Wilcoxon signed rank test, $p=0.25$). However, when the inactive participants were removed (resulting in N=65), a significant improvement was visible ($p<0.05$). Since most participants started out in a stage that corresponded to good performance (A or M, N=56), another analysis was performed that focused on the participants that were in a stage in which they could improve, i.e. in stages PC, P, or C. Participants who started in stage PC, P or C (N=18) improved significantly over the course of the study (Wilcoxon signed rank test, $p<0.01$). On average they improved one stage. Furthermore, the targeted constructs on average significantly increased compared to the non-targeted constructs (t-test, $p<0.001$). This change was particularly large for the constructs susceptibility, skills and emotions.

4 Discussion & Future Work

There are a few limitations to the presented study. First, to draw conclusions about an enduring change, behavior should be measured over a longer period of time (e.g., 6 months). Thus, it should be stressed that the aim of this study was not to evaluate the long-term effectiveness of a behavior change intervention, but rather to evaluate the reasoning mechanism of eMate in a domain in which behavior can be influenced relatively quickly. Secondly, there was no control group in the experiment. Given this design, we cannot make conclusive statements about the effectiveness of the intervention. Furthermore, many challenges for future work remain open. We will concentrate on evaluating eMate in other domains and developing parameter tuning methods to find optimal parameter values to use the model for prediction.

Acknowledgments. This research was supported by Philips and Technology Foundation STW, Nationaal Initiatief Hersenen en Cognitie NIHC under the Partnership programme Healthy Lifestyle Solutions.

References

[1] B. A. Kamphorst, M. Klein, and A. van Wissen. Autonomous e-coaching in the wild: Empirical validation of a model-based reasoning system. In *Proceedings of the 13h International Conference on Autonomous Agents and Multiagent Systems (AAMAS '14)*, pages 725–732, 2014.

[2] M. Klein, N. Mogles, and A. van Wissen. Why won't you do what's good for you? Using intelligent support for behavior change. In *International Workshop on Human Behavior Understanding (HBU'11)*, volume 7065 of *LNCS*, pages 104–116. Springer Verlag, 2011.

Market Garden: a Simulation Environment for Research and User Experience in Smart Grids[1]

Bart Liefers [a] Felix N. Claessen [a] Eric Pauwels [a]

Peter A.N. Bosman [a] Han La Poutré [ab]

[a] *Centrum Wiskunde & Informatica (CWI), Amsterdam, The Netherlands*
[b] *TUDelft, Delft, The Netherlands*

Abstract

Market Garden is a scalable research environment and demonstration tool, in which market mechanisms for smart energy systems and the interaction between end users, traders and system operators can be simulated. Users can create scenarios in a user-friendly editor in which a hierarchical market architecture and a model of the physical grid can be defined. The system sets up an interactive and competitive environment of modular market mechanisms. A visualiser in which the results can be explored in a graphical and intuitive way is included, making Market Garden very suitable for user experience and demonstration purposes, next to its application for research.

1 Introduction

In electrical power systems, a transition towards a more dynamic, decentralised, and interactive structure takes place. Especially in and around the distribution grid, many innovations occur such as smart houses, electric vehicles, micro-CHP and distributed renewables with uncertain generation (wind, solar). To cope with such an increasingly complex system, more of the final energy demand should follow the (uncertain) sustainable energy supply, and new pricing and allocation mechanisms are needed for continuously matching supply and demand. One of the desirable properties of such mechanisms is the ability to make commitments ahead of time in order to ensure a reliable energy supply. Planning ahead is important for e.g. power generators that face startup time, or electric vehicles that need to be fully charged in time. Other desirable properties include efficiency, fairness, transparency and scalability.

To develop appropriate allocation mechanisms that have these properties, the interplay between different actors needs to be studied. Currently, several simulation tools are available that focus on a specific mechanism within the energy system, such as retail or wholesale, with other mechanisms considered fixed. However, it is interesting to see how changes in one mechanism lead to a response in, for example, the strategies of agents or in regulation.

In this paper we present Market Garden, a simulation environment for research and user experience in smart grids. Market Garden has a distinctive combination of important features compared to other multi-agent systems (MASCEM[1], Power TAC[2]) used for simulations in the energy domain: 1) Market Garden both includes links between end users, retail, wholesale trading and the distribution grid, and allows developers to test and validate agents that represent any of these actors. The interactive and competitive environment set up by Market Garden allows researcher to study the interplay between, for example, novel market mechanisms, agents' trading strategies and regulatory actions. 2) The market mechanisms in Market Garden are modular; a library of basic market mechanisms is included, and developers can easily develop and add new mechanisms. 3) The included mechanisms have parameterised

[1] The full paper has been published in *Advances in Practical Applications of Heterogeneous Multi-Agent Systems: proceedings of the 12th PAAMS conference*, Springer LNAI 8473, pages 351–354, 2014. Supported by EIT ICT Labs.

time schedules, allowing users to specify market open and close horizons, timeslot durations, and repetition intervals, and making it possible to simulate trade on arbitrary time scales rather than within fixed timeslots. The time between trading and delivery of electricity can significantly influence the behaviour of traders in a specific market, making timing an important subtopic of research in market design. 4) A model of the physical grid underlies the multi-agent system for energy trade, which allows us to include transmission losses and ancillary services. 5) The inherent modularity and hierarchical representation of the physical network ensures scalable and parallel computation.

2 The system

Market Garden includes several standard implementations of actors involved in energy trading, such as wholesale market operators, retailers, end users and Distribution System Operators (DSOs). These actors are represented in the system as modules, which can be expanded or replaced entirely.

A DSO monitors the energy flows through the physical network and resolves any imbalances that may occur by efficiently dispatching ancillary services. End users (prosumers) can trade energy through e.g. a retailer or a wholesale market, and are represented as nodes in the physical network. Retailers interact with end users by offering tariffs, energy prices to which end users can subscribe, which can vary on arbitrary time scales (hourly, weekly, monthly). A retailer is responsible for balancing the energy prosumption of its customers, and can trade on wholesale markets to achieve this. Any imbalance costs are assigned to the retailer through the balancing market, which is present in any simulation. Additionally, multiple wholesale markets (day-ahead, intra-day) can be defined, on which futures (binding commitments for energy in future timeslots) can be traded.

Market results, energy prosumption, network flows, and market organisation can all be explored in a graphical and intuitive way in the software's visualiser (Figure 1). This makes Market Garden very suitable for user experience and demonstration purposes towards end users, traders, or e.g. DSOs in a smart grid environment, in addition to simulations for research purposes.

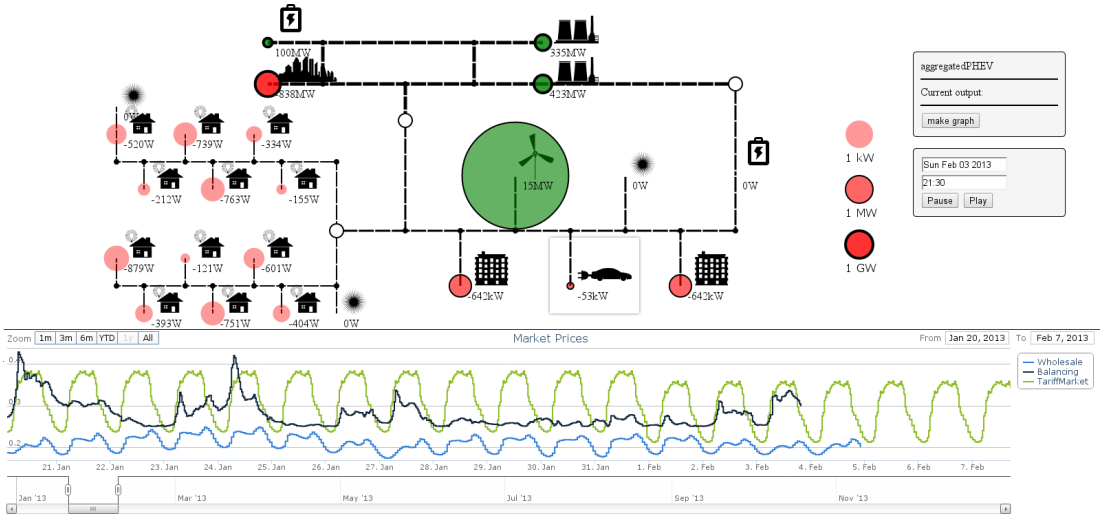


Figure 1: Network representation of the visualiser, with green circles for energy production at nodes and red for consumption. The chart at the bottom depicts the fluctuation of market prices (green, blue and black lines represent unit prices of a time-of-use tariff, wholesale, and balancing power, respectively).

References

[1] Praca, I., Ramos, C., Vale, Z., and Cordeiro, M.: MASCEM: A Multiagent System that Simulates Competitive Electricity Markets. IEEE Intelligent Systems 18(6), 54-60 (2003)

[2] Ketter, W., Collins, J., and Reddy, P.: Power TAC: A Competitive Economic Simulation of the Smart Grid. Energy Economics 39, 262-270 (2013)

Measuring Diversity of Preferences in a Group[1]

Vahid Hashemi Ulle Endriss

Institute for Logic, Language and Computation (ILLC)
University of Amsterdam

Abstract

We introduce a general framework for measuring the degree of diversity in the preferences held by the members of a group. We formalise and investigate three specific approaches within that framework: diversity as the range of distinct views held, diversity as aggregate distance between individual views, and diversity as distance of the group's views to a single compromise view. While similarly attractive from an intuitive point of view, the three approaches display significant differences when analysed using both the axiomatic method and empirical studies.

1 Introduction

Preferences are ubiquitous in AI [2]. Examples for application domains include recommender systems, planning, and configuration. Of particular interest is the case of preference handling in multiagent systems, where several agents each have their own individual preferences and we need to take decisions that are appropriate in view of such a profile of preferences. The normative, mathematical and algorithmic aspects of this problem are studied in the field of (computational) social choice [1].

Intuitively speaking, one might expect that the less *diverse* the preferences in a group of agents are, the easier it should be to come to mutually acceptable decisions. For example, in the most extreme case where all agents share the exact same preference order, it will be trivial to make collective decisions. *Vice versa*, the more diversity we find in a group, the more we should expect to encounter paradoxes, i.e., situations in which different social choice-theoretic principles would lead to opposing conclusions. In recent work we have proposed a new formal model of preference diversity to study such phenomena [3]. In this extended abstract we sketch this formal model and report on some of the results obtained.

2 Preference Diversity Indices

Let $\mathcal{X}$ be a finite set of m *alternatives*. We model *preferences* as (strict) *linear orders* over $\mathcal{X}$ and write $\mathcal{L}(\mathcal{X})$ for the set of such all preference orders. Let $\mathcal{N} = \{1, \ldots, n\}$ be a finite set of *agents*. A *profile* $\boldsymbol{R} = (R_1, \ldots, R_n) \in \mathcal{L}(\mathcal{X})^n$ is a vector of preference orders, one for each agent. The *support* of a profile $\boldsymbol{R} = (R_1, \ldots, R_n)$ is the set of preference orders occurring in it: $\textsc{Supp}(\boldsymbol{R}) = \{R_1\} \cup \cdots \cup \{R_n\}$. We call a profile $\boldsymbol{R}$ *unanimous* if $|\textsc{Supp}(\boldsymbol{R})| = 1$, i.e., if it is of the form $(R, \ldots, R)$.

Given two profiles $\boldsymbol{R}$ and $\boldsymbol{R'}$ (with the same number of agents n expressing preferences over the same number of alternatives m), we want to be able to make judgments about which of them we consider to be more diverse. To this end, we introduce the concept of *preference diversity index*.

Definition 1. *A **preference diversity index** (PDI) is a function $\Delta : \mathcal{L}(\mathcal{X})^n \to \mathbb{R}^+ \cup \{0\}$, mapping profiles to the nonnegative reals, such that $\Delta(\boldsymbol{R}) = 0$ for any unanimous profile $\boldsymbol{R} \in \mathcal{L}(\mathcal{X})^n$.*

[1] The full paper appears in the *Proceedings of the 21st European Conference on Artificial Intelligence (ECAI-2014)*.

Given a particular PDI Δ, we say that profile $\boldsymbol{R}$ is more diverse than profile $\boldsymbol{R'}$ if $\Delta(\boldsymbol{R}) > \Delta(\boldsymbol{R'})$. That is, any given PDI Δ induces a weak order on the set of all profiles (which we call a *preference diversity order*), ranking them from most to least diverse. We now introduce three specific families of PDI's:

(1) The *simple support-based PDI* counts the number of distinct preference orders in a profile. This idea can be generalised to counting, for a given $k \leqslant m$, the number of distinct ordered k-tuples of alternatives appearing in a profile (suitably normalised to ensure unanimous profiles map to 0).

(2) Under a *distance-based PDI*, we measure the distance (e.g., the *Kendall tau distance* or the *discrete distance*) between any two preference orders in a given profile and then aggregate the values obtained (e.g., by computing their sum or their maximum).

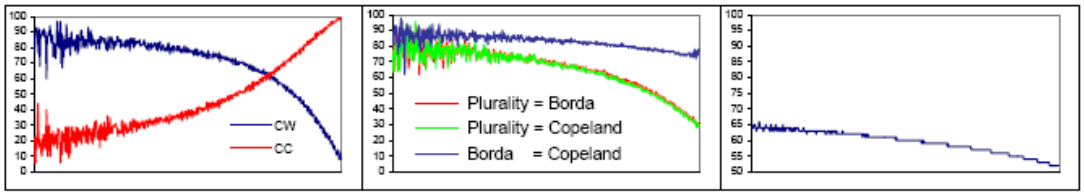
(3) Under a *compromise-based PDI*, we first aggregate the individual preferences (e.g., using the *Borda rule* or the *majority rule*), then compute the Kendall tau distance of each individual preference to that "compromise", and finally aggregate (e.g., add) the values thus obtained.

Other instantiations of the concept of PDI are certainly possible, and we consider identifying further specific PDI's an important direction for future work.

In the full paper we propose several *axioms* capturing certain desirable properties of PDI's and analyse which specific PDI's satisfy which of these properties [3].

3 Experimental Results

We have conducted several experiments, using both synthetic preference data and profiles sampled from real preference data, to obtain a better understanding of the concept of PDI, and in particular to gain insights into the effects of diversity on certain social choice-theoretic phenomena. As an example, consider the three graphs below, which concern scenarios with 5 alternatives and 50 voters (averaged over 1 million profiles drawn from the synthetic distribution defined by the so-called *impartial culture assumption*, under which every logically possible preference profile is equally likely to occur in practice):



For all three graphs, the x-axis corresponds to the diversity of profiles as measured by the distance-based PDI computing the sum of Kendall tau distances between pairs of individual preference orders. On the y-axis we see the frequency (in percent) of a particular phenomenon occurring. The first graph shows that, as diversity increases, the chance of encountering a profile with a *Condorcet winner* decreases, while the chance of encountering a profile with a *Condorcet cycle* increases. The second graph plots the frequency of two voting rules agreeing on the election winner against the diversity of the preferences of the voters. The third graph shows how average voter satisfaction (measured in terms of the proportion of alternatives a voter ranks above the election winner, in this case determined using the Borda rule) decreases as diversity increases.

Thus, we see a clear correlation between increases in diversity and both decreases of desirable effects and increases of undesirable effects. These results match our intuition well, thereby giving credence to our claim that PDI's are a useful formal model for capturing the important but elusive notion of diversity.

References

[1] F. Brandt, V. Conitzer, and U. Endriss. Computational social choice. In G. Weiss, editor, *Multiagent Systems*, pages 213–283. MIT Press, 2013.

[2] C. Domshlak, E. Hüllermeier, S. Kaci, and H. Prade. Preferences in AI: An overview. *Artificial Intelligence*, 175(7):1037–1052, 2011.

[3] V. Hashemi and U. Endriss. Measuring diversity of preferences in a group. In *Proceedings of the 21st European Conference on Artificial Intelligence (ECAI-2014)*. IOS Press, 2014.

Multi-objective Gene-pool Optimal Mixing Evolutionary Algorithms[1]

Ngoc Hoang Luong Han La Poutré Peter A.N. Bosman

Centrum Wiskunde & Informatica (CWI), Amsterdam, The Netherlands

1 Introduction

Multi-objective (MO) optimization problems deal with multiple conflicting objectives at the same time. Each solution represents a trade-off between different objectives, and a utopian solution optimizing all objectives is non-existent. Instead of searching for one best solution as in single-objective (SO) optimization, MO optimizers try to obtain a good approximation $\mathcal{S}$ of the true Pareto-optimal front $\mathcal{P}_F$ of the problem at hand, which consists of multiple non-dominated solutions. The goal of MO optimization has thus two aspects: proximity (i.e. the approximation set $\mathcal{S}$ should be as close as possible to $\mathcal{P}_F$) and diversity (i.e. set $\mathcal{S}$ should be well-spread along $\mathcal{P}_F$). It has been showed that MO evolutionary algorithms (MOEAs) are well-suited and, indeed, an effective methodology for solving MO optimization problems. However, while classical MOEAs are effective in achieving this two-fold goal, the issue of scalability, which depends on the algorithms' capability for linkage relation learning, is often overlooked. MO estimation-of-distribution algorithms (MOEDAs), such as the Multi-objective Adapted Maximum-Likelihood Model (MAMaLGaM) or the Multi-objective Hierarchical Bayesian Optimization Algorithm (mohBOA), tackle the scalability issue by replacing classic variations operators with model-based variation operators.

In this paper, by constructing the Multi-objective Gene-pool Optimal Mixing Evolutionary Algorithm (MO-GOMEA), we pinpoint key features for scalable MO optimizers. First, an elitist archive is beneficial for keeping track of non-dominated solutions. Second, clustering can be crucial if different parts of the Pareto-optimal front $\mathcal{P}_F$ need to be handled separately. Next, an efficient linkage learning procedure with a lean linkage model is required to capture the underlying dependencies among decision variables. It is also important that the optimizers can effectively exploit the learned linkage relations to generate new offspring solutions, steering the search toward promising regions in the search space.

2 Key Features

Elitist Archive[1]. The number of Pareto-optimal solutions can be numerous or even infinite, exceeding the limited capacity of the population. Furthermore, some good solutions can be lost due to the stochastic nature of the selection and variation operators. It is thus beneficial to have an archive keeping track of the non-dominated solutions found so far, and, in turn, the current Pareto front constituted by these solutions. The criteria for accepting a newly generated solution $\boldsymbol{x}$ into the archive $\mathcal{A}$ are that $\boldsymbol{x}$ is not dominated by any existing members of $\mathcal{A}$ and that $\boldsymbol{x}$ can change the Pareto front found so far. If $\boldsymbol{x}$ can be added to $\mathcal{A}$, solutions dominated by $\boldsymbol{x}$ are removed from $\mathcal{A}$. Mechanisms to maintain the diversity or to adapt the archive when its capacity is exceeded can also be employed.

Clustering [1]. In every generation, the population of candidate solutions is partitioned into k equal-sized clusters. Each cluster can be seen as being allocated an equal amount of resource for approaching a different part of the Pareto-optimal front $\mathcal{P}_F$. The extreme solutions (i.e. the one that is optimal in terms of a single objective) and their corresponding extreme regions on $\mathcal{P}_F$ can be very inefficient

[1]The full paper has been accepted for presentation at *The Genetic and Evolutionary Computation Conference (GECCO 2014)*.

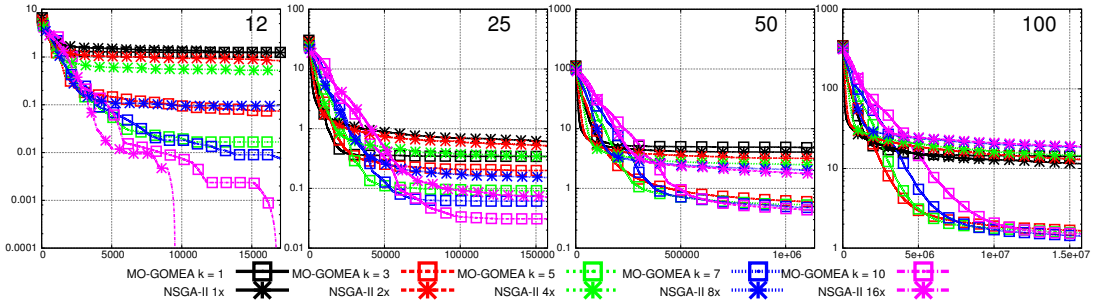


Figure 1: Performance of MO-GOMEA and NSGA-II on multi-objective MAXCUT problems of different problem sizes. Horizontal axis: number of evaluations. Vertical axis: $\boldsymbol{D}_{\boldsymbol{\mathcal{P}_F}\rightarrow\mathcal{S}}$

for MO optimization to obtain. It is thus more beneficial if these extreme regions are handled by SO optimization. Therefore, for every objective, the cluster having the best cluster mean value in that objective is assigned to perform SO optimization operations regarding that objective.

Linkage Learning [2]. After partitioning the population, a set $\boldsymbol{\mathcal{S}}_i$ of solutions having better fitness values is selected from each cluster $\boldsymbol{\mathcal{C}}_i$. A linkage model $\mathcal{F}_i$ for cluster $\boldsymbol{\mathcal{C}}_i$ is then learned from this selection set $\boldsymbol{\mathcal{S}}_i$. MO-GOMEA employs the Linkage Tree (LT) structure as its linkage model. The LT contains linkage groups representing the dependencies among decision variables, and organizes them in a tree-like fashion. Leaf nodes of LT are univariate linkage groups, capturing decision variables as being fully independent. Branch nodes of LT are multivariate linkage groups, indicating that the decision variables in each group have some degree of dependency and thus should be treated jointly together when performing variation. The LT can be learned by an efficient hierarchical clustering procedure.

Optimal Mixing (OM) [2]. From each existing solution $\boldsymbol{x}$, the OM operator uses the LT $\mathcal{F}_i$ of the cluster $\boldsymbol{\mathcal{C}}_i$, to which $\boldsymbol{x}$ belongs, to construct a new offspring solution $\boldsymbol{o}$ by iteratively improving $\boldsymbol{x}$. First, $\boldsymbol{o}$ is cloned directly from $\boldsymbol{x}$. Looping through the LT $\mathcal{F}$, for every linkage group $\boldsymbol{F} \in \mathcal{F}$, a random donor solution $\boldsymbol{p}$ is selected from the same cluster $\boldsymbol{\mathcal{C}}_i$. The values of the decision variables indicated by $\boldsymbol{F}$ are copied from $\boldsymbol{p}$ to $\boldsymbol{o}$. If such mixing create a new (partial) solution $\boldsymbol{o}'$ which dominates $\boldsymbol{o}$ or is a non-dominated solution regarding the archive $\boldsymbol{\mathcal{A}}$, then $\boldsymbol{o}'$ is accepted as the new $\boldsymbol{o}$. Otherwise, $\boldsymbol{o}$ is rolled back to the previous state. Note that for solutions $\boldsymbol{x}$ belonging to extreme clusters, OM improves $\boldsymbol{x}$ in SO optimization manner.

3 Experiment & Discussion

We show the performance of our MO-GOMEA with the well-known NSGA-II on solving the MO-MAXCUT. Results of other benchmarks can be found in the full paper. We experiment with instances of MO-GOMEA having different number of cluster $k = 1, 3, 5, 7, 10$. Multiple instances of NSGA-II are created by scaling the base population size to 1,2,4,8, and 16 times. We employ the inverse generational distance $\boldsymbol{D}_{\boldsymbol{\mathcal{P}_F}\rightarrow\mathcal{S}}$ as the performance indicator. Fig. 1 shows convergence graphs of $\boldsymbol{D}_{\boldsymbol{\mathcal{P}_F}\rightarrow\mathcal{S}}$ from the beginning until termination, averaged over 100 independent runs. For MAXCUT instance having problem size $l = 12$, only NSGA-II $16\times$, the one with the largest population size, has better performance than MO-GOMEA. However, as the problem size increases ($l \geq 25$), MO-GOMEA starts to outperform NSGA-II. The bigger the problem is, the wider the performance gap between MO-GOMEA and NSGA-II becomes, indicating the superior scalability of MO-GOMEA over NSGA-II. Note that MO-GOMEA without clustering ($k = 1$) has performance results relatively the same to those of NSGA-II instances, which do not have clustering either. This emphasizes the importance of clustering to handle different parts of the Pareto-optimal front separately. Without linkage learning, NSGA-II cannot solve problems of large problem sizes as efficiently as MO-GOMEA even though it is equipped with very large population sizes (NSGA-II $8\times$ or $16\times$). Having a linkage learning mechanism and effectively exploiting the linkage model are thus crucial for a scalable MO optimizer.

References

[1] P. A. N. Bosman. The anticipated mean shift and cluster registration in mixture-based edas for multi-objective optimization. In *Proc. of the Genetic and Evolutionary Comp. Conf. - GECCO-2010*, pages 351–358.

[2] P. A. N. Bosman and D. Thierens. More concise and robust linkage learning by filtering and combining linkage hierarchies. In *Proc. of the Genetic and Evolutionary Comp. Conf. - GECCO-2013*, pages 359–366.

A Successful Broker Agent for Power TAC[1]

Bart Liefers Jasper Hoogland Han La Poutré

CWI Amsterdam

Abstract

The Power TAC simulates a smart grid energy market. In this simulation, broker agents compete for customers on a tariff market and trade energy on a wholesale market. It provides a platform for testing strategies of broker agents against other strategies. In this paper we describe the strategies of our broker agent. Amongst others, due to a beneficial trading technique related to equilibria in continuous auctions on the wholesale market and a strategy inspired by Tit-for-Tat in the Iterated Prisoner's Dilemma game on the tariff market, our broker ended second in the 2013 Power TAC.

1 Introduction

The Power Trading Agent Competition (Power TAC) is a platform in which agent strategies for trading energy in a smart grid setting can be tested [1]. Competing researchers create retailer agents (referred to as 'brokers') whose goal is to maximize their profit. They do this by publishing tariffs to attract customers and by trading on a wholesale market.

In this paper we present cwiBroker, a broker that was created for the Power TAC in 2013. In the tariff market, our broker distinguishes between duopoly games (games with two brokers) and oligopoly games (games with four or seven brokers). In duopoly games, our broker uses an adaptive strategy inspired by Tit-for-Tat in the Iterated Prisoner's Dilemma game [2]. In oligopoly games our broker aims to find the optimal tariff by estimating the profit for a set of candidate tariffs. Finally, for the wholesale market we used a beneficial trading technique, related to equilibria in continuous auctions. Below, we state our approach.

2 Duopoly Tariff Strategy

In order to obtain a high revenue on the tariff market in duopoly games, brokers must have a high tariff price, while at the same time have a large number of customers. As a result of the competition for customers, however, brokers may be forced to decrease their tariff price, which will have a negative impact on their revenue. It would be better for both brokers to avoid competition, by allowing each other to have high tariff prices. This "cooperation" enables both brokers to have a large number of customers and a high tariff price at the same time. However, this approach can only be successful if both brokers cooperate. A cooperative broker performs poorly if the other broker is competitive, because its expensive tariffs will not attract any customers. A competitive broker, on the other hand, would never benefit from the other broker's willingness to cooperate. We would like to have a broker that cooperates with the other broker if possible, but competes for customers if necessary. We achieved this by implementing a tariff strategy inspired by the Tit-for-Tat strategy in the Iterated Prisoner's Dilemma [2] game.

In more detail, our broker publishes an initial tariff with a price p_{init} (chosen manually). After that, whenever the opponent publishes a tariff with price p', our broker responds by replacing its current tariff with a new tariff with price $p = p' - \Delta$. First we choose $\Delta > 0$, but after two times we use $\Delta = 0$. At this stage our strategy resembles Tit-for-Tat, as it now copies the other broker's tariff. Our broker is competitive if the other broker is competitive, and it is cooperative if the other broker is cooperative.

[1]The full paper has been published in *Proceedings of the Joint Workshop on Trading Agent Design and Analysis (TADA) and Agent-Mediated Electronic Commerce (AMEC)*, 2014.

3 Oligopoly Tariff Strategy

In oligopoly games we expect the competition among brokers to be much higher than in duopoly games. Therefore, a cooperative approach is unlikely to succeed. Due to the high competition, though, tariff prices will quickly stabilize close to the cost price. Our broker estimates a subscription model $\hat{N}(p)$, which yields for any tariff price p the number of customers that will subscribe to such a tariff. This model is an adjusted and parametrized version of the tariff evaluation used by customers in the simulation [1]. The parameters of the model are computed from the number of customers subscribed to two test tariffs, which are published, one at the time, during the early stages of the game. The subscription model is used to estimate the profit for a set of candidate tariffs. Our broker publishes the tariff that it expects to be the most profitable. Even though the number of customers depends on the other brokers' tariffs, we expect it can be estimated successfully, due to the stability of the tariff prices in oligopoly games. In case the estimated optimal tariff does not attract enough customers, our broker switches to an adaptive strategy. It will then decrease its tariff price, until a minimal number of customers is obtained.

4 Wholesale Strategy

On the wholesale market, electricity is traded in a double-sided sequential auction. Due to the predictable behaviour of sellers, we know the sell orders in each auction quite accurately. Furthermore, we can estimate the total energy demanded by all brokers. This information determines the equilibrium price, at which all brokers can buy their demand in the first round of the sequential auction. However, other brokers were expected to bid lower than the equilibrium price, at the price of the cheapest sell orders, in an attempt to obtain this cheap energy. Therefore, our broker bids higher than the other brokers in the first auction round, but still lower than the equilibrium price. Furthermore, it did not just bid for its own demand Q_{cwi}, but for the total demand Q_{total} of all brokers. This way, it acquired energy Q_1 at a price lower than the equilibrium price, before several other brokers acquired their energy. In case $Q_1 > Q_{cwi}$, our broker sold the quantity $Q_1 - Q_{cwi}$, which it did not need itself, to the other brokers in the remaining auction rounds. The broker could offer it at a price lower than the cheapest remaining sell order, so it was guaranteed to sell it, but still higher than the price at which it was bought.

5 Results

Our broker ended second out of seven participants in the competition. It obtained the highest score in each of the 7-player games and in five of the six 2-player games it played.

In duopoly games, our broker was usually forced to be competitive, but in most of the games it was able to get customers, since it adapted accordingly. In duopoly games with one particular broker, cooperation proved very beneficial. Though we obtained fewer customers than some competitive brokers in these games, we obtained the highest profit. In oligopoly games, our broker overestimated the estimated optimal tariff, but thanks to the adaptive strategy it was always able to attract some customers. The wholesale strategy resulted in the lowest average wholesale price of all brokers.

6 Summary and Conclusion

We developed cwiBroker for the Power TAC in 2013. The combined set of strategies for the tariff market and the wholesale market resulted in a versatile and robust broker, that performed very well in all settings that were tested.

References

[1] Ketter, W., Collins, J., Reddy, P., and de Weerdt, M.: The 2013 Power Trading Agent Competition, ERIM Report Series Reference No. ERS-2013-006-LIS, 2013

[2] Axelrod, R., and Hamilton,W.D.: The Evolution of Cooperation, Science 27 March 1981: 211 (4489), 1390-1396

Smarter smartphones: understanding and predicting user habits from GPS sensor data*

Giuseppe Maggiore[a†] Carlos Santos[a] Aske Plaat[a,b]

[a] *Academy for Digital Entertainment, NHTV Univ. of Applied Sciences, Breda*
[b] *Leiden Institute of Advanced Computer Science, Leiden University*

Abstract

Smartphones and similar devices allow access to a wealth of information. Semantic locations can help navigating this wealth of information by providing information that is useful to the user at a specific time or place, without the need for the user to perform an explicit query. This paper shows how a stream of sensor data can be processed and interpreted to determine (i) the locations of interest for a user, such as home, work, etc, and (ii) to predict future transitions between such locations. We have implemented our algorithms in a fully functional prototype smartphone app and backend, and we present results based on actual usage data gathered in early 2013. Our system is open source, and can be used to build context-aware recommender systems that suggest content which is at the right time and at the right place.

1 Introduction

Smartphones and similarly "always on, always connected" devices allow access to a wealth of information [4]. Public transportation time-tables, shops with sales, new restaurants, public events, etc. can be accessed through these devices. This is one of the major contributing factors of the widespread adoption of mobile devices in modern society [2]. To make smartphone even more useful, two drawbacks must be solved: *(i)* the amount of available information is overwhelming, and is not all relevant for each user; and *(ii)* small devices can be cumbersome to type on for extended periods, often making navigation of this wealth of information problematic.

In the paper we discuss a working implementation of how observation of user movements (via GPS), together with an Internet connection, allows inferencing semantic locations [3] such as home, favourite shops and restaurants, gyms, parks, workplace, etc. Such knowledge is used to offer personalized, contextual information without the need for the user to perform a query; for example, the device can vibrate when passing in front of a shop, displaying the message "this shop now has a sale, you may want to check it!".

The paper is concerned with the following problem statement: *How can the stream of sensor data gathered from a smartphone can be automatically processed in order to determine: (i) the* ***locations of interest*** *of a user and their semantic label such as home, work, etc; and (ii) the* ***predicted future transitions*** *between such locations?* We present a working implementation of our algorithms, and an evaluation.

2 The idea

The location inference system is based on *density-based clustering*. The shape (see Figure 1) of clusters is determined by the average velocity of the samples associated with the cluster. We distinguish samples

*The full paper has been published in *Proceedings of the 11th International Conference on Mobile Systems and Pervasive Computing (MobiSPC-2014)*

†giuseppemag@gmail.com

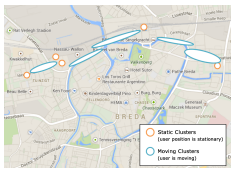


Figure 1: Moving and static clusters

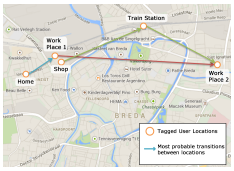


Figure 2: Some entries from the transitions tables; color of arrow represents length of transition

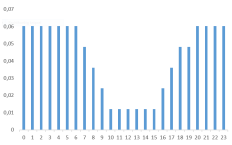


Figure 3: Home hits by hour

in two groups: *(i)* static, the user was *not moving*; and *(ii)* dynamic, the user was *moving*. Clusters that contain static entries have a circular shape defined by the maximum radius allowed for clusters. Moving clusters are shaped like an ellipse, where the ellipse axis is parallel to the average velocity of the entries recorded in the cluster.

Clusters "degrade" over time. If a cluster is not updated regularly, then it is removed from the system. This ensures that only fresh information is maintained.

We use the history of visited locations to infer a Markov process. The nodes of the process are the clusters found at the previous step of the algorithm, and the transitions are (intuitively enough) the transitions between clusters that were witnessed so far. For an example, see Figure 2. Transitions also have a *duration*. The duration of a transition measures how long it took to travel between the clusters that this transition connected. For each supported transition duration, we store a separate table. The supported transition durations grow exponentially.

The Markov process is also used for semantic label inference. We build, off-line, a series of probability distributions for each possible label. These distributions describe the expected patterns for assigning labels to a cluster. For example, we expect that the user will spend late afternoons and nights at home, and so the probability distribution of expected home usage will look like Figure 3. We compare the expected distributions with those we observed with the Kullback-Leibler (KL) [1] distance between probability distributions.

3 Conclusions and future work

In the paper, we show how the stream of sensor data gathered from a smartphone can be automatically processed in order to determine: *a*) the **locations of interest** of a user and their semantic label such as *home*, *work*, etc. *b*) the **expected future transitions** between such locations. We have tested our prototype on a group of students, obtaining encouraging results. Our system iss able to correctly identify their locations of interest and their movements. Our system shows that a layered combination of smeantic location techniques allows powerful inference about users, and offers useful techniques for improved location-aware recommendations.

References

[1] S. Kullback and R. A. Leibler. On information and sufficiency. *Annals of Mathematical Statistics*, 1951.

[2] N.D. Lane, E. Miluzzo, Hong Lu, D. Peebles, T. Choudhury, and A.T. Campbell. A survey of mobile phone sensing. *Communications Magazine, IEEE*, 2010.

[3] Salil Pradhan. Semantic location. *Personal Technologies*, 2000.

[4] Dan Wang, Park Sangwon, and Fesenmaier Daniel. An examination of information services and smartphone applications. 2011.

Efficient Heuristics for Power Constrained Planning of Thermostatically Controlled Loads[1]

Frits de Nijs Mathijs M. de Weerdt Matthijs T. J. Spaan

Delft University of Technology
{f.denijs, m.m.deweerdt, m.t.j.spaan}@tudelft.nl

1 Introduction

Supply and demand in an electricity grid need to match in real-time to ensure stability. Production from fossil fuel generators is controllable, and can be made to follow the (predicted) demand. However, as more and more generation is replaced by uncontrollable sources such as wind and solar, the supply side will become less flexible. In order to handle supply anomalies, the demand will need to respond. The ideal demand side response is storage: stored over-supply can later compensate for under-supply.

A large potential storage capacity can be found in the various heat buffers operated by consumers: houses, refrigerators and hot water reservoirs all need to be maintained at a certain temperature. All of these buffers decay to the environment temperature over time, which requires constant action to counteract. But precisely when these buffers are heated or cooled can be shifted in time. By storing more energy in the heat buffer now, we can later 'extract' this heat from the buffer in the form of reduced loads. In this sense, we may think of heat buffers controlled by thermostats as a kind of batteries [1]. Because the duration of supply shortages is potentially long enough to violate the consumers comfort constraints [2], we propose using planning to respond to fluctuations before they occur, minimizing the discomfort experienced by the end-users.

2 Model and Problem

A Markov chain model of heat buffers controlled by thermostats was presented by Mortensen and Haggerty [3]. In their model, the temperature θ evolves under influence from the insulation level a, the current outside temperature θ_t^{out}, and a temperature input from the device θ^{pwr} activated through binary control action m_t , giving in the model: $\theta_{t+1} = a\theta_t + (1-a)\left(\theta_t^{\text{out}} + m_t\theta^{\text{pwr}}\right)$. Given the effective power consumption P^{eff}, we can determine the load at time t to be $m_t P^{\text{eff}}$.

Using boldface characters to represent vectors of device parameters, we propose an optimization problem to minimize a discomfort cost function $c(\theta)$ over a finite horizon h, given a power consumption constraint L_t (using Hadamard product $\boldsymbol{a} \circ \boldsymbol{b} \implies a_i \times b_i \ \forall i$):

$$\begin{aligned} \underset{[\,\boldsymbol{m}_0\ \boldsymbol{m}_1\ \cdots\ \boldsymbol{m}_h\,]}{\text{minimize}} \ & \sum_{t=0}^{h} c(\boldsymbol{\theta}_t) \\ \text{subject to } & \boldsymbol{\theta}_{t+1} = \boldsymbol{a} \circ \boldsymbol{\theta}_t + (1-\boldsymbol{a}) \circ \left(\theta_t^{\text{out}} + \boldsymbol{m}_t \circ \boldsymbol{\theta}^{\text{pwr}}\right) \\ & \boldsymbol{m}_t \cdot \boldsymbol{P}^{\text{eff}} \leq L_t \\ & m_{i,t} \in [0,1] \qquad \forall i,t \end{aligned} \tag{1}$$

[1] The full paper has been published in *International Workshop on Demand Response*, ACM e-Energy, 2014.

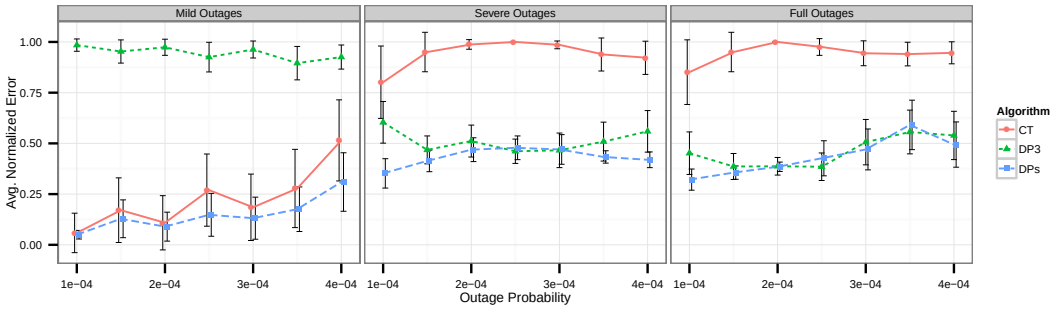


Figure 1: Performance of the heuristic under varying levels of power constraints.

3 Proposed Heuristic Solution

The optimization problem can be solved using the standard backward recursive dynamic programming algorithm, but that will quickly become intractable since the state space $\boldsymbol{\theta}_t$ and the action space $\boldsymbol{m}_t$ are both exponential in n (proof of NP-completeness in full paper). Instead, we propose to solve the optimization problem for a single device with averages $\bar{a}_i$ and $\bar{\theta}^{\text{pwr}}$. Such an average device requires a different transition function, which takes into account the power constraint L_t.

We estimate the maximum next average temperature from L_t by computing the maximum average duty cycle $D = \frac{L_t}{P^{\max}}$. Given this duty cycle, the maximum temperature can be approximated from a period of heating of length D and a period of cooling of length $(1 - D)$.

When executing the plan produced for the average device, we give priority to the devices with the largest error. We assume devices can estimate their error contribution through the difference between the measured temperature and the set-point. We sort the devices by their estimated reduction in error, and switch on as many as remain feasible, starting from the highest error reduction.

4 Evaluation

In the experimental evaluation, we consider houses of varying build quality by simulating 500 houses with randomized a and θ^{pwr} under increasing power constraints: during 'Mild' outages, approximately two-thirds of the average unrestricted load remains available, while during 'Severe' outages at most one-third of the needed power is available. During 'Full' outages no power is available. The frequency of the outages is controlled through the probability of an outage occurring on each time step.

Figure 1 presents the resulting normalized error: The method with the highest error is assigned a normalized error of 1, and the others are then computed through their ratio to the worst method. The results are averaged over 20 runs per setting. A single run takes less than 10 seconds to plan. As the severity of the outages increases, the benefit of planning becomes larger compared to just using control (CT). But when the outages become very frequent, errors start becoming unavoidable which translates into an increasing error for increasing outage frequency for the DP heuristics. The three-state planning algorithm (DP3) does not perform well under 'Mild' outages, which is a consequence of the continuous error incurred from staying just inside the error-free band 'on average'. The more fine-grained offset-based algorithm (DPs) instead consistently outperforms the control mechanism.

References

[1] H. Hao, B. M. Sanandaji, K. Poolla, and T. L. Vincent. "A Generalized Battery Model of a Collection of Thermostatically Controlled Loads for Providing Ancillary Service". In: *Annual Allerton Conference on Communication, Control and Computing*. 2013.

[2] S. Koch, M. Zima, and G. Andersson. "Active Coordination of Thermal Household Appliances for Load Management Purposes". In: *Symposium on Power Plants and Power Systems Control*. 2009, pp. 149–154.

[3] R. E. Mortensen and K. P. Haggerty. "A stochastic computer model for heating and cooling loads". In: *IEEE Transactions on Power Systems* 3.3 (1988), pp. 1213–1219.

Nash Equilibria in Shared Effort Games*

Gleb Polevoy [a] Stojan Trajanovski [a] Mathijs de Weerdt [a]

[a] *Delft University of Technology, Mekelweg 4, 2628 CD, Delft, The Netherlands.*

Abstract

Shared effort games model people's contribution to projects and sharing the obtained profits. Those games generalize both public projects like writing for Wikipedia, where everybody shares the resulting benefits, and all-pay auctions such as contests and political campaigns, where only the winner obtains a profit. In θ-equal sharing (effort) games, a threshold for effort defines which contributors win and then receive their (equal) share. (For public projects $\theta = 0$ and for all-pay auctions $\theta = 1$.) Thresholds between 0 and 1 can model games such as paper co-authorship and shared homework assignments. We study existence and efficiency of such games, to know what will happen in a given situation and where an intervention may be needed to improve the social welfare. First, we fully characterize the conditions for the existence of a pure-strategy Nash equilibrium for two-player shared effort games with close budgets and project value functions that are linear on the received contribution and prove some efficiency results. Second, since the theory does not work for more players, fictitious play simulations are used to show when such an equilibrium exists and what its efficiency is. The results about existence and efficiency of these equilibria provide the likely strategy profiles and the socially preferred strategies to use in real life situations of contribution to public projects.

1 Introduction

People often invest in projects and share their revenues. Some examples are contributions to Wikipedia, or paper co-authorship [3]. In several real-life situations, like bonus points from shared homeworks, only the contributors who contribute at least a given threshold of the maximum contribution receive some profit. We want to know when such a situation will settle in an efficient equilibrium, and when some intervention may be required.

To model the situation we have described, we now define *shared effort games* (also appeared in [1]) that consist of a set of players and a set of projects. Each player has a budget to split somehow between a predefined subset of projects. The utility of each player is the sum of the utilities that she obtains from each project. A project's value is a function of the investments of the players in a given project. We consider a specific variant of a shared effort game, called a *θ-sharing mechanism*. In this mechanism, the project's value is equally divided between all the users who contribute at least θ of the maximum bid to the project.

Contributing to projects where only the maximum contributor to a project obtains the project's value, has been considered in all-pay auctions. The efficiency was bounded in [1], but assuming some very specific conditions. However, there is no analysis of the existence of Nash Equilibria (NE) and their efficiency in the general effort sharing interaction. Therefore, we analyze the existence and the efficiency of equilibria, to recommend socially optimum behavior to the contributors. We first theoretically characterize existence of NE in a particular case and find its efficiency. To consider some other case, we generalize fictitious play[1] to shared effort games (which are infinite), and use them to find NE. To simulate this, we provide a best response algorithm. We consider only pure NE throughout the paper.

*The full paper has been published in *Proceedings of the 13th International Conference on Autonomous Agents and Multiagent Systems*, pages 861–868, 2014.

[1] The original fictitious play was proposed by Brown [2].

2 Theory of Nash Equilibrium

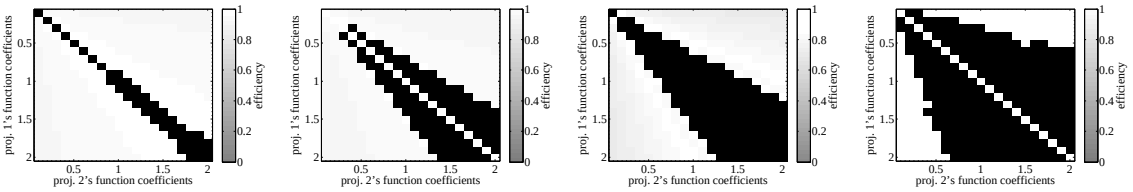
We study the existence of NE, and when it exists, we consider its efficiency. First, we prove some existence results for continuous sharing of projects' utilities. For the case of two players with close budgets and linear project functions, we characterize the existence of a NE. For a general n, some sufficiency conditions can be proven.

Since there may be various Nash Equilibria, we consider the price of anarchy (PoA), which is the ratio of the worst NE's efficiency to the optimum possible efficiency, and the price of stability (PoS), which is the ratio of the best NE's efficiency to the optimum possible efficiency. We prove that in the case that a NE exists in the above characterization, then almost always $\mathrm{PoA} = \mathrm{PoS} = 1$.

3 Simulations and Conclusions

To study when a game possesses at least one NE for some cases that are not yet handled, we generalize the fictitious play, originally suggested by Brown [2] for mixed extensions of finite games, and simulate it to find a NE. To simulate this, we devise an algorithm to find a best response to a given opponents' profile, if it exists. Any convergence provides a candidate for a NE, which is checked. Sometimes, our simulations find a NE for some games but we never assert that no NE exists.

Some results for the θ-equal 2-project case are presented in Fig. 1.



(a) $B_1 = 1$, $B_2 = 10$, $\theta = 0.2$ (b) $B_1 = 1$, $B_2 = 10$, $\theta = 0.5$ (c) $B_1 = 2$, $B_2 = 5$, $B_3 = 15$, $\theta = 0.5$ (d) $B_1 = 2$, $B_2 = 2$, $B_3 = 4$, $\theta = 0.2$

Figure 1: The simulation results for various thresholds θ and budgets B_i. Bold black color means that Nash Equilibrium has not been found. For all the other cases, the efficiency, a value in $[0, 1]$, is shown by the shade (from gray = 0.0 to white = 1.0).

For two players with far away budgets, a NE exists when the project function coefficients are not too close to one another. For $\theta = 0.5$, also equal coefficients bring upon a NE. For three players, a NE exists except, perhaps, when the project functions are quite close to each other.

The efficiency is close to optimum, except for the case of far away budgets, when the efficiency drops from somewhere in the interval $[0.6, 0.8]$ when the project function coefficients are quite close, to somewhere in the interval $[0.53, 0.68]$ when the project functions differ the most.

To summarize, the paper studied existence and efficiency of the NE in shared effort games, characterizing it for two players with close budgets, where the project functions are linear, and looking for NE through generalized fictitious play simulations for 2 projects. The found NE are optimum in almost all considered cases, besides in the case of three players with far away budgets.

Thus, this paper implies that for three or more players, some regulation may improve the total utility.

References

[1] Y. Bachrach, V. Syrgkanis, and M. Vojnović. Efficiency and the Redistribution of Welfare. Technical report, Microsoft Research, October 2011.

[2] George W Brown. Iterative solution of games by fictitious play. *Activity analysis of production and allocation*, 13(1):374–376, 1951.

[3] Jon Kleinberg and Sigal Oren. Mechanisms for (mis)allocating scientific credit. In *Proceedings of the 43rd annual ACM symposium on Theory of computing*, pages 529–538, New York, NY, USA, 2011. ACM.

A Novel Population-based Multi-Objective CMA-ES and the Impact of Different Constraint Handling Techniques[1]

Sílvio Rodrigues [a] Pavol Bauer [a] Peter A.N. Bosman [b]

[a] *Delft University of Technology (TU Delft), Delft, The Netherlands*
[b] *Centrum Wiskunde & Informatica (CWI), Amsterdam, The Netherlands*

1 Introduction

Many real-world problems have more than one conflicting objective that need to be optimized simultaneously. Moreover, many problems require taking a black-box optimization (BBO) perspective, i.e. assume that (virtually) nothing is known (e.g. complex simulation-based real-world models). Studying and understanding algorithms to tackle optimization problems under such conditions is therefore important.

The Covariance Matrix Adaptation Evolutionary Strategy (CMA-ES) is a well-known, state-of-the-art optimization algorithm for single-objective real-valued (BBO) problems. Although several extensions of CMA-ES to multi-objective (MO) optimization exist, none incorporates a key component of the most robust CMA-ES variant: associating a population with each Gaussian that drives optimization.

Many real-world problems also have constraints, making the performance of BBO algorithms under different constraint-handling techniques important. All MO-CMA-ES variants previously introduced use a penalty term to handle box constraints. Although this resulted in fast convergence speeds for certain benchmark problems, it also has drawbacks. For example, it only performs well with box constraints since these, differently from general problem constraints, allow an easy mapping to the feasible space. Furthermore, infeasible solutions may end up in the elitist archive.

The main objectives of this paper stem from the fact that all existing MO-CMA-ES variants use populations of size one and that only the penalty approach was used to handle constraints. Our first goal is to study the benefits of having a population-based MO-CMA-ES. To do so, we study a combination between a multi-objective optimization framework that was recently introduced [1] with the most general SO version of CMA-ES [2]. Our second goal is to assess the performance and robustness of the previously introduced MO-CMA-ES variants, the novel population-based MO-CMA-ES and the iMAMaLGaM algorithm [1] under different and more general constraint handling techniques.

2 Population-based Multi-Objective CMA-ES

A general framework for extending population-based algorithms from single- to multi-objective optimization was introduced in [1]. We made a few improvements to this framework and used it to construct a population-based multi-objective version of CMA-ES. For the minor improvements, we refer the reader to the full paper. Here we only give a high-level flavor of the workings of the framework. In addition to common domination-rank-based selection, variation is ensured to be based on clustering. That is, the selected solutions are explicitly clustered in the objective space and each cluster undergoes variation separately. An important part of state-of-the-art variation operators are adaptive mechanisms that span multiple generations. The performance of these mechanisms strongly depends on a correlation between the solution sets in subsequent generations. Therefore, some form of registration is required to

[1] The full paper has been accepted for presentation at The Genetic and Evolutionary Computation Conference (GECCO 2014).

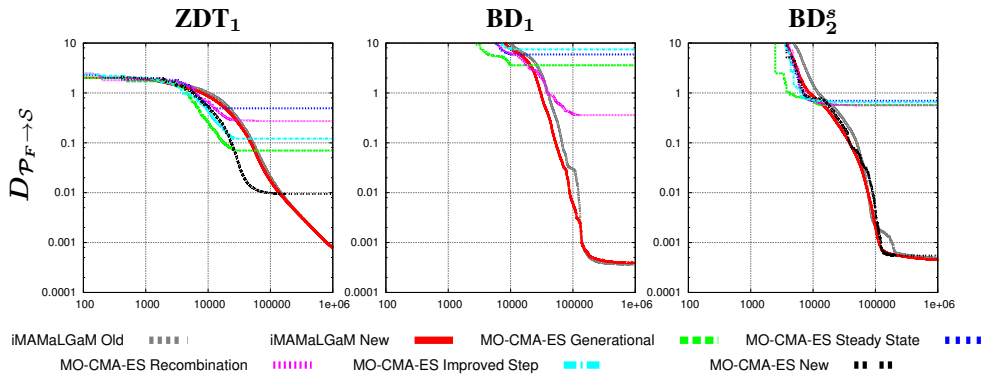


Figure 1: Performance of all algorithms on selected problems, averaged over 100 runs. Horizontal: number of evaluations. Vertical: distance-to-optimum indicator (lower is better, 0.001 is value to reach).

determine the best correspondence between clusters in subsequent generations, for which a greedy algorithm is used to construct the best pairing of clusters in subsequent generations. Every cluster is then allowed to generate an equal number of solutions through variation whereby for each objective one cluster is identified that focuses solely on making improvements in that objective so as to specifically add pressure on extending the Pareto front along a single axis. Finally, an elitist archive is maintained with all currently non-dominated solutions. If the objectives are real valued, infinitely many non-dominated solutions are possible. To prevent the archive from growing to extreme sizes, the objective space is adaptively discretized into hypercubes so as to accommodate a predefined desired number of solutions.

3 Results

The results are averaged over 100 independent runs and shown for the most interesting cases in Figure 1 (for more graphs, see the full version of this paper). The differences between the old and new versions of iMAMaLGaM have a positive, but small effect in terms of convergence.

No MO-CMA-ES from literature could solve unconstrained problem BD_2^s. Only one of the Pareto extremes was found because BD_2^s has one objective that is far simpler than the other, resulting in the population being pulled quickly toward one end of the Pareto front. Due to the lack of pressure toward improving individual objectives as in the MO framework, it is hard for existing MO-CMA-ES implementations to find the other Pareto extreme, resulting in very low convergence speed and early stopping the optimization runs. Population-based MO-CMA-ES and the iMAMaLGaM variants did solve BD_2^s.

The other problems do have constraints, making them harder, especially for CMA-ES. Although the population-based variant of CMA-ES always did better than existing CMA-ES variants, no variant of CMA-ES could solve the constrained problems satisfactorily using any tested constraint-handling technique. This is due to the careful way CMA-ES has been designed. When more non-smooth search-space adaptations occur, for instance as a result of constraint handling, the powerful search-space exploitation capacity of CMA-ES breaks down. In this regard, the iAMaLGaM basis appears to be more robust as its multi-objective counterpart could solve all problems with all constraint handling techniques.

4 Conclusions

We introduced a novel population-based MO-CMA-ES. Experimental results demonstrate that the proposed approach is, in general, more robust when compared to the multi-objective extensions of CMA-ES that were previously introduced in literature. Furthermore, the algorithms based on CMA-ES demonstrated to be very sensitive to the way constraints were handled. By comparison, iMAMaLGaM showed to be more robust since it was able to solve all problems with all constraint handling techniques.

References

[1] Peter A. N. Bosman and T. Alderliesten. Incremental Gaussian model-building in multi-objective edas with an application to deformable image registration. In *Proceedings of the Genetic and Evolutionary Computation Conference - GECCO-2012*, pages 241–248, New York, 2012. ACM Press.

[2] N. Hansen and S. Kern. Evaluating the CMA evolution strategy on multimodal test functions. In X. Yao et al., editors, *Parallel Problem Solving from Nature (PPSN VIII)*, pages 282–291. Springer, 2004.

Bounded Approximations for Linear Multi-Objective Planning under Uncertainty

(Extended Abstract)

Diederik M. Roijers [a] Joris Scharpff [b] Matthijs T.J. Spaan [a]

Frans A. Oliehoek [a] Mathijs M. De Weerdt [b] Shimon Whiteson [a]

{d.m.roijers, f.a.oliehoek, s.a.whiteson}@uva.nl
[a] *University of Amsterdam, The Netherlands*
{j.c.d.scharpff, m.t.j.spaan, m.m.deweerdt}@tudelft.nl
[b] *Delft University of Technology, The Netherlands*

Abstract

Planning under uncertainty poses a complex problem in which multiple objectives often need to be balanced. When dealing with multiple objectives, it is often assumed that the relative importance of the objectives is known a priori. However, in practice human decision makers often find it hard to specify such preferences exactly, and would prefer a decision support system that presents a range of possible alternatives. We propose two algorithms for computing these alternatives for the case of linearly weighted objectives. First, we propose an anytime method, approximate optimistic linear support (AOLS), that incrementally builds up a complete set of ϵ-optimal plans, exploiting the piecewise-linear and convex shape of the value function. Second, we propose an approximate anytime method, scalarised sample incremental improvement (SSII), that employs weight sampling to focus on the most interesting regions in weight space, as suggested by a prior over preferences. We show empirically that our methods are able to produce (near-)optimal alternative sets orders of magnitude faster than existing techniques, thereby demonstrating that our methods provide sensible approximations in stochastic multi-objective domains.

1 Introduction

Many real-world planning problems involve both uncertainty as well as multiple objectives. This type of problems is expressed naturally using the multi-objective Markov decision process (MOMDP) framework [4]. Following [4] we assume the existence of a *scalarisation function*, i.e. a function that translates multi-dimensional rewards into a scalar value. However, using such a function for planning requires complete knowledge of its parameters, or *weights*, beforehand. When such knowledge is not available, solving an MOMDP requires finding the set of optimal solutions for all possible weights.

In this paper[1], we consider only linear scalarisation functions. Therefore, it suffices to focus on the *Convex Coverage Set* of an MOMDP. Existing methods such as optimistic linear support (OLS) [5] exploit the value piecewise-linear convexity in the optimal value function over all weights, present when the scalarisation function is linear, to minimise the number of scalarised MDPs that need to be solved. However, OLS can only guarantee this when the scalarised MDPs are solved optimally and is therefore not directly applicable to large realistic planning problems.

2 Our Contributions

We propose new methods that rely on approximate MDP solving techniques to produce (near-)optimal CCSs. The first algorithm we propose is approximate optimistic linear support (AOLS) that, given an

[1] This is an extended abstract of our paper [3] at ICAPS 2014.

ϵ-bounded MDP approximation, is guaranteed to produce an ϵ-approximate CCS. The second algorithm, scalarised sample-based iterative improvement (SSII), exploits available prior knowledge on the distribution of weights and concentrates its effort within such a *prior*.[2] Although SSII can in practice produce a better approximate CCS over this prior, we cannot provide a bound on the CCS quality because SSII relies on sampling.

Both AOLS and SSII use an approximate single-objective solver as a subroutine. AOLS can use any solver. SSII requires an anytime method. In this paper we use UCT* [2] for both.

3 Evaluation

We performed experiments on instances of the maintenance planning problem [6], a 2-objective, probabilistic and numerical planning domain, and compared the optimal OLS method with our approximate AOLS and SSII. The optimal solutions have been computed using SPUDD [1] and approximations using the UCT* algorithm from PROST [2]. We compared the outcomes in terms of runtime, average CCS error ε_{exp} and maximal CCS ε_{max} error. The results are presented in Table 1 below.

			$[0,1]$			$[0.5,1]$		
Algorithm	Runtime	$\lvert CCS\rvert$	ε_{exp}	ε_{max}	%OPT	ε_{exp}	ε_{max}	%OPT
OLS + SPUDD	2390.819	9.250	-	-	-	-	-	-
AOLS + UCT* 0.01s	8.612	3.389	0.701	325.354	0.000	0.692	325.025	0.000
AOLS + UCT* 1s	19.940	4.111	0.119	65.668	0.167	0.117	65.426	0.167
AOLS + UCT* 10s	65.478	4.528	0.084	56.439	0.333	0.091	56.381	0.333
AOLS + UCT* 20s	165.873	5.694	0.044	38.667	0.417	0.048	38.627	0.417
SSII 1s, no prior	18.795	4.306	0.118	70.244	0.167	0.116	70.195	0.167
SSII 10s, no prior	59.336	3.889	0.061	51.800	0.333	0.068	51.747	0.333
SSII 1s, prior	17.892	3.944	0.221	95.189	0.000	0.125	61.667	0.167
SSII 10s, prior	59.154	4.083	0.141	71.290	0.083	0.057	43.006	0.333

Table 1: *Comparison of averaged performance of the algorithms presented in this paper for various parameters, shown for two regions of the scalarised reward space. Runtimes are in seconds, the expected error ε_{exp} and maximum error ε_{max} are relative to the optimum CCS and %OPT denotes the fraction of instances that were solved optimally.*

From the table we can conclude that both AOLS and SSII are able to produce reasonable, and sometimes even optimal, solutions much faster than OLS. Also, SSII is competitive with AOLS without exploiting additional knowledge but when SSII uses the prior it produces a slightly better CCS within the targeted weight region $w_1 \in [0.5, 1]$ and $w_2 = 1 - w_1$.

References

[1] Jesse Hoey, Robert St-Aubin, Alan Hu, and Craig Boutilier. SPUDD: Stochastic Planning Using Decision Diagrams. In *Proc. of the Fifteenth conference on Uncertainty in artificial intelligence*, pages 279–288, 1999.

[2] Thomas Keller and Malte Helmert. Trial-based Heuristic Tree Search for Finite Horizon MDPs. In *Proc. of the International Conference on Automated Planning and Scheduling*, 2013.

[3] Diederik M. Roijers, Joris Scharpff, Matthijs T.J. Spaan, Frans A. Oliehoek, Mathijs de Weerdt, and Shimon Whiteson. Bounded approximations for linear multi-objective planning under uncertainty. In *Proceedings of the 24th International Conference on Automated Planning and Scheduling (ICAPS)*, pages 262–270, June 2014.

[4] Diederik M. Roijers, Peter Vamplew, Shimon Whiteson, and Richard Dazeley. A Survey of Multi-Objective Sequential Decision-Making. *Journal of Artificial Intelligence Research*, 47:67–113, 2013.

[5] Diederik M. Roijers, Shimon Whiteson, and Frans Oliehoek. Linear Support for Multi-Objective Coordination Graphs. In *Proc. of the Autonomous Agents and Multi-Agent Systems conference*, 2014.

[6] Joris Scharpff, Matthijs T. J. Spaan, Leentje Volker, and Mathijs de Weerdt. Planning Under Uncertainty for Coordinating Infrastructural Maintenance. *Proc. of the International Conference on Automated Planning and Scheduling*, 2013.

[2] Note that such a prior expresses some — but not complete — knowledge about the weights.

Combining Simulated Annealing and Monte Carlo Tree Search for Expression Simplification[1]

Ben Ruijl [ab] Jos Vermaseren [a] Aske Plaat [b] Jaap van den Herik [b]

[a] *Nikhef, Science Park 105, 1098 XG Amsterdam, The Netherlands*
[b] *Leiden University, Niels Bohrweg 1, 2333 CA Leiden, The Netherlands*

Abstract

In this work we present a modification to UCT. Previously, C_p was a constant that determined the exploration-exploitation trade-off; in the new version C_p is linearly decreased with the iteration number. For the problem of expression simplification with Horner schemes, this change results in a lower sensitivity to the C_p parameter by at least a factor of 10.

1 Expression simplification

In High Energy Physics (HEP), expressions with millions of terms arise from the calculation of processes described by Feynman diagrams. Typically, these expressions have to be numerically integrated to predict cross sections and particle decays in collision processes. For example, in the Large Hadron Collider in CERN such calculations were essential to confirm the likely existence of the Higgs boson. For future predictions, the expressions will be exponentially larger, so we need a novel approach to simplify expressions.

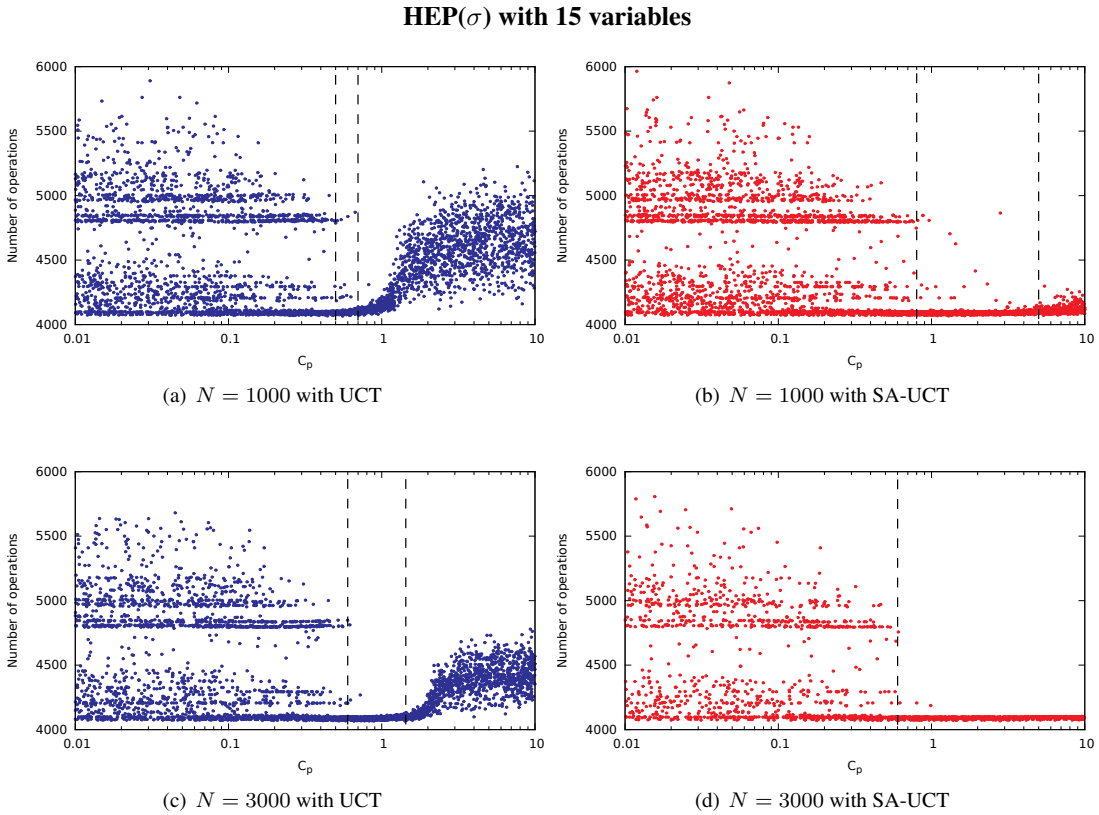
Horner schemes and common subexpression elimination (CSEE) may be the best means to simplify expressions. Horner's rule is lifting variables outside brackets, in order to reduce the number of multiplications. On a multivariate polynomial, the order in which the variables are extracted is called a Horner scheme. Finding the Horner scheme that yields the best reduction is an NP-hard problem. Common subexpression elimination removes subexpressions that appear more than once, and thus reduces both the number of multiplications and additions. In this work we continue research on finding Horner schemes for which the number of operations is lowest after both the scheme and CSEE have been applied.

2 SA-UCT

In previous work, Monte Carlo Tree Search (MCTS) has been applied for finding Horner schemes. UCT (Upper Confidence bounds applied to Trees) was used as the selection criterion. The performance of MCTS is sensitive to the choice of three parameters: C_p, N, and R. C_p is the constant in the UCT formula that governs the exploration-exploitation choices of the algorithm, N is the number of tree updates, and R is the number of times MCTS is repeated. At the previous BNAIC conference the sensitivity to C_p and N was presented.

This paper focuses on the role of the C_p. We modify the UCT formula by introducing an exploration-exploitation parameter $T(i)$, which decreases with the current iteration number i, effectively making the constant C_p a variable $T(i)$. As a result, the first iterations will be explorative and throughout the search,

[1] The full paper has been published in *Proceedings of the International Conference on Agents and Artificial Intelligence*, pages 724–731, 2014.



(a) $N = 1000$ with UCT

(b) $N = 1000$ with SA-UCT

(c) $N = 3000$ with UCT

(d) $N = 3000$ with SA-UCT

Figure 1: The number of operations is on the y-axis and C_p on the x-axis. A lower number of operations is better. N indicates the number of iterations. On the left, we show UCT where C_p is constant, and on the right we show SA-UCT where C_p is the starting value of T. Each graph contains 4000 runs (dots) of MCTS. As indicated by the dashed lines, an area with an operation count close to the global minimum appears, as soon as there are sufficient tree updates N. This area is wider for SA-UCT than for UCT.

the child selection will gradually become more exploitative, favoring optimizing a local minimum over exploration. The parameter T is similar to the role of the temperature in simulated annealing (hence the name T). We refer to the new formula as Simulated Annealing UCT (SA-UCT).

We have performed a sensitivity analysis of C_p and N for SA-UCT for seven large expressions from mathematics and physics, of which three are worked out in detail in the full paper. In Figure 1 we show the result for the HEP(σ) expression. We observe that for both UCT and SA-UCT with $N = 1000$, there are three regions: an exploitation-dominated low C_p region where the simulations converge to local minima, an intermediate region where almost all simulations find the global minimum (indicated by dashed lines), and a cloud-like exploration-dominated region at high C_p, where there is no convergence. We observe that for SA-UCT with $N = 1000$, the region of good results, i.e., the intermediate region, is at least ten times larger compared to the region for UCT. For $N = 3000$, we observe that the successful region for SA-UCT ranges from 0.6 to (over) 10. In passing we see that for UCT with $N = 3000$ the region is also increased, but not as much as for SA-UCT.

We may conclude that by using SA-UCT the tuning problem has been significantly simplified. As a result, it will take less time to optimize expressions from High Energy Physics. Using SA-UCT, expressions can be simplified with at least a factor 20. The consequences are immediate: numerical evaluations that used to take several weeks can now be done in days.

Combining Model-Based EAs for Mixed-Integer Problems[1]

Krzysztof L. Sadowski [a] Dirk Thierens [a] Peter A. N. Bosman [b]

[a] *Utrecht University, The Netherlands*
[b] *CWI Amsterdam, The Netherlands*

1 Introduction

A key characteristic of Mixed-Integer (MI) problems is the presence of both continuous and discrete problem variables. In this paper, we study the design of an algorithm that integrates the strengths of LTGA [2] [3] and iAMaLGaM [1]: state-of-the-art model-building Evolutionary Algorithms (EAs) designed for discrete and continuous search spaces, respectively. We wish to study if making use of the model building and learning abilities of both these algorithms can be applied to MI problems while retaining excellent scale-up behavior. The model-building nature of these algorithms allows us to consider black-box problems where no prior information about a problem structure is known. How difficult is it to achieve a proper evaluation balance and adequate scalability as the problem size increases? Is it even possible to solve dependent problems where continuous variables interact with the discrete ones, while using integrated but independently learning models?

2 Problems

We use well-established problems (see full paper for definitions) and adapt them into the MI setting. We consider different combinations of discrete and continuous problems where the contributions of the discrete and continuous parts are kept independent through addition. Minimization is assumed.

In the full paper, we introduce five functions, F_1 - F_5 which represent different types of variable dependencies. Variables in F_1 are fully independent. Only continuous variables are dependent in F_2. Only discrete variables are dependent in F_3. In F_4 both sub-spaces are dependent. The F_5 benchmark includes cross-domain dependencies between the continuous and discrete variables. It is an additively decomposable specific combination of the deceptive trap function with the rotated ellipsoid (see full paper for details).

3 Results

For the analysis of $F_1 - F_4$, we consider different problem lengths l. For each problem size, we consider different proportions of variables used with 5, $0.25l$, $0.5l$, $0.75l$ and $l-5$ continuous variables (remaining variables are discrete). Success criterion is solving a problem 29/30 times with the precision of 10^{-10}.

Heat maps in Figure 1 show that more evaluations are required for the same problem sizes as the composition of the problem shifts towards more continuous variables. Moreover, benchmarks which contain dependencies within the continuous sub-space, F_2 and F_4, require larger number of evaluations than F_1 or F_3. Population sizes are also affected by the problem composition. In F_3 and F_4 we observe much larger population size requirements, as the landscape of these functions includes discrete

[1] The full paper has been accepted for publication in the *13th International Conference on Parallel Problem Solving from Nature (PPSN'14)*."

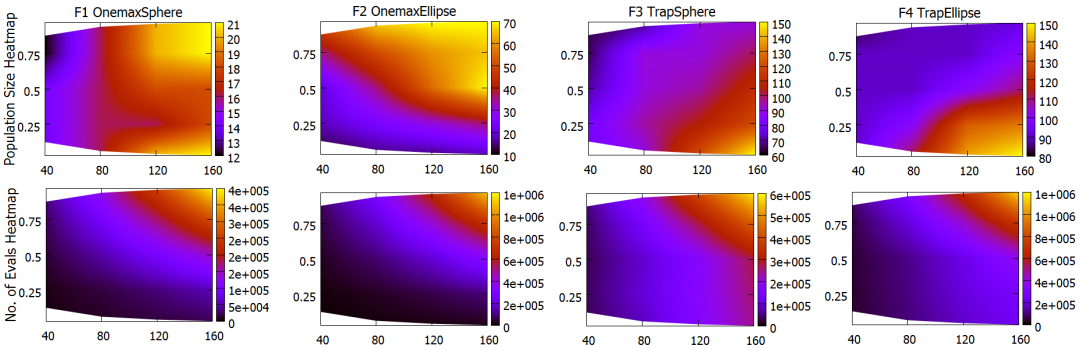


Figure 1: Heat Maps representing the population sizes (top row) and evaluations (bottom row) needed for different variable compositions. Horizontal axis represents the problem length, the vertical axis is the fraction of continuous variables ($l_c/(l_c + l_d)$) in the problem

variable dependencies. This shows that in addition to problem length, the composition of the problem and variable dependencies are a big factor for efficiency in terms of evaluations and population sizes. Scalability analysis in the full paper verified that the results exhibit polynomial scalability on the tested MI problems. Results on F_5 in the full paper show that the hybrid algorithm we propose is in fact capable of solving this dependent benchmark, however not in all cases. When the fitness contribution of the discrete variables is scaled down to very small values, it causes their initial fitness contributions to be very small, resulting in the algorithm prematurely converging on sub-optimal solutions. As this contribution is scaled up, the problem becomes simpler and requires smaller population sizes and less evaluations in order to be solved.

4 Conclusions

Mixed-Integer problems introduce many optimization challenges which do not arise in purely real or discrete optimization problems. Obtaining a proper balance in exploration of model information for different types of variables, varying variable ratios and additional overhead or fitness contribution scaling are some of the important issues which should be taken into account when solving MI problems. Our algorithm achieved polynomial scale-up behavior on the tested benchmarks. We showed that a well-balanced algorithm can solve some cases of even very dependent mixed-integer problems, despite having independent model learning methods for the discrete and continuous sub-spaces. The results provide a good foundation and motivation for further work in mixed-integer landscapes with model building EAs.

References

[1] Peter A. N. Bosman, Jörn Grahl, and Dirk Thierens. Enhancing the Performance of Maximum-Likelihood Gaussian EDAs Using Anticipated Mean Shift. In *Parallel Problem Solving from Nature — PPSN X*, volume 5199 of *LNCS*, pages 133–143, 2008.

[2] Dirk Thierens. The linkage tree genetic algorithm. In *Parallel Problem Solving from Nature — PPSN XI*, volume 6238 of *LNCS*, pages 264–273, Berlin, 2010. Springer–Verlag.

[3] Dirk Thierens and Peter A N Bosman. Optimal mixing evolutionary algorithms. In *Proceedings of the Genetic and Evolutionary Computation Conference*, GECCO '11, pages 617–624, New York, NY, USA, 2011. ACM.

Anchor-Profiles: Exploiting Profiles of Anchor Similarities for Ontology Mapping

Frederik C. Schadd [a] Nico Roos [a]

[a] *Department of Knowledge Engineering, Maastricht University, The Netherlands*

Abstract

We present a novel approach for the exploitation of partial alignments for the task of ontology mapping. The correspondences of a partial alignment are referred to as anchors. The core idea is that if two ontology concepts are similar, then their comparison with the anchor concepts of a provided partial alignment is likely to produce the same results. To facilitate this comparison, we create an anchor-profile for each concept, such that the profile contains the result of all similarity evaluations between the associated concept and the given anchors. Our experiments demonstrate good results supporting further research.

1 Introduction

Ontology mapping is a necessary task for resolution of ontology heterogeneity, facilitating the exchange of information between two knowledge system which utilize different ontologies. Such a situation can occur in the case of a business acquisition or business co-operation, where one business needs access to the knowledge system of another business. Traditionally, ontology mappings are generated by a domain expert, however the size of ontologies in contemporary knowledge systems have made this solution infeasible. As a result, ontology mapping systems are developed with the aim of automatically generating these mappings.

For some tasks, a partial alignment might be available, which can be the result of a domain expert having insufficient time to make a complete mapping. Such a partial alignment can be exploited as additional input source in order to produce more accurate mappings [2]. In this paper [4], we present a novel approach for the mapping of ontologies using partial alignments.

2 Anchor Profiles

For partial alignment, the general assumption is that each anchor is assumed to be correct. Otherwise, special techniques can be applied to improve their reliability [5]. Assuming the correctness of the anchors, the main intuition behind our approach is that two concepts can be considered similar if they exhibit a comparable degree of similarity towards a given anchor. More formally, given two ontologies O_1 and O_2, and given an anchor $A_x[C^1, C^2]$ containing a correspondence between the concepts C^1 and C^2 originating from O_1 and O_2 respectively, and given a concept similarity $sim'(E, F) \in [0, 1]$ which expresses the similarity between two concepts, we define an anchor similarity $sim_A(C, A_x)$ between an arbitrary concept C and A_x as:

$$sim_A(C, A_x) = \begin{cases} sim'(C, C^2) & \text{if } C \in O_1 \\ sim'(C, C^1) & \text{if } C \in O_2 \end{cases} \quad (1)$$

From equation 1 follows that two concepts C and D can be considered similar if $sim_A(C, A_x)$ and $sim_A(D, A_x)$ are similar. Given that a partial alignment most likely contains multiple correspondences, this intuition needs to be expanded for a series of anchors. For this purpose, we define the

Anchor-Profile of a concept C as a vector $Profile(C) = (sim_A(C, A_1), \ldots, sim_A(C, A_x))$, where the given partial alignment $PA = \{A_1, \ldots, A_x\}$ is defined as a set of anchors. The similarity between profiles is computed using a vector similarity sim_P, for which the cosine similarity was chosen for this research. Thus, we compute the anchor-profile similarity between C and D by computing $sim_P(Profile(C), Profile(D))$.

3 Evaluation

We evaluate our approach on the *biblio-benchmark* dataset from the Ontology Alignment Evaluation Initiative [1]. We generate the input partial alignments by randomly sampling correspondences from the provided reference alignment, with each task being evaluated 100 times. To remove the bias introduced by the partial alignment, which is also included in the output alignment, we apply adapted measures of precision, recall and f-measure, P^*, R^* and F^* respectively, which do not include the anchors in their computation [4]. s'_{P^*}, s'_{R^*} and s'_{F^*} denote the pooled standard deviations of P^*, R^* and F^*. As sim' we applied an aggregate of similarities which were obtained from the *MaasMatch* mapping system [3]. The evaluation was repeated for various sizes of the generated partial alignment, denoted as *PA Recall*.

PA Recall	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
P^*	0.760	0.769	0.779	0.786	0.801	0.817	0.835	0.855	0.866
R^*	0.632	0.641	0.649	0.656	0.663	0.674	0.685	0.702	0.745
F^*	0.668	0.678	0.686	0.693	0.701	0.713	0.726	0.743	0.780
s'_{P^*}	0.094	0.099	0.107	0.112	0.125	0.139	0.155	0.180	0.219
s'_{R^*}	0.049	0.068	0.083	0.092	0.102	0.117	0.133	0.158	0.215
s'_{F^*}	0.038	0.053	0.066	0.074	0.083	0.098	0.115	0.142	0.199

Table 1: Results of the evaluations on the benchmark-biblio dataset using different recall requirements for the randomly generated partial alignments.

From Table 1, we can observe resulting adapted precisions in the interval $[0.76, 0.87]$, adapted recalls in $[0.63, 0.75]$ and adapted f-measures in $[0.66, 0.78]$. Furthermore, we observe that all three measures gradually rise when increasing the size of PA. We observe a moderate instability of these measure for large partial alignments, which likely stems from the fact that any variation in the produced alignment will have a larger impact on P^*, R^* and F^* if PA has a significant size.

Furthermore, a comparison with other mapping systems has been performed which evaluated the same data-set in the OAEI competition. From this comparison we observed that the quality of the produced alignments using the anchor-profile approach is on par with the top mapping systems that participated in the competition, especially when using large partial alignments.

References

[1] Aguirre, J., Grau, B., Eckert, K., Euzenat, J., Ferrara, A., van Hague, R., Hollink, L., Jimenez-Ruiz, E., Meilicke, C., Nikolov, A., Ritze, D., Shvaiko, P., Svab-Zamazal, O., Trojahn, C., Zapilko, B.: Results of the ontology alignment evaluation initiative 2012. In: Proc. of OM2012. pp. 73–115 (2012)

[2] Noy, N.F., Musen, M.A.: Anchor-prompt: Using non-local context for semantic matching. In: Proceedings of the ICJAI workshop on ontologies and information sharing. pp. 63–70 (2001)

[3] Schadd, F.C., Roos, N.: Maasmatch results for oaei 2012. In: Proceedings of The Seventh ISWC International Workshop on Ontology Matching. pp. 160–167 (2012)

[4] Schadd, F.C., Roos, N.: Anchor-profiles for ontology mapping with partial alignments. In: Proceedings of the 12th Scandinavian AI Conference (SCAI 2013). pp. 235–244. IOS (2013)

[5] Schadd, F.C., Roos, N.: A feature selection approach for anchor evaluation in ontology mapping. In: Proceedings of the 5th International Conference on Knowledge Engineering and Semantic Web (KESW) (2014), (Accepted Paper)

Causal discovery from databases with discrete and continuous variables[1]

Elena Sokolova [a] Perry Groot [a] Tom Claassen [a] Tom Heskes[a]

[a] *Radboud University, Faculty of Science,*
Postbus 9010, 6500 GL Nijmegen, The Netherlands

1 Introduction

Causal discovery is widely used for analysis of experimental data focusing on the exploratory analysis and suggesting probable causal dependencies. There is a variety of causal discovery algorithms in the literature. Some of these algorithms rely on the assumption that there are no latent variables in the model; others do not provide a scoring metric to easily compare the reliability of two candidate models. Bayesian Constraint-based Causal Discovery (BCCD) [1] is a state-of-the-art-algorithm for causal discovery that tries to combine the strength of the best algorithms in the field. BCCD is able to detect latent variables in the model and determines the reliability of the edges between variables that makes it very easy to compare alternative models.

The idea of BCCD is to estimate the reliability of causal relations by scoring Directed Acyclic Graphs (DAGs) for a smaller subset of variables using a Bayesian score and then to combine these statements to infer a final causal model. The Bayesian score has a closed form solution for discrete variables that makes the scoring of causal relations fast and efficient. The BCCD algorithm is currently limited to discrete or Gaussian variables as there is no closed form solution for the Bayesian score for a mixture of discrete and continuous variables. To extend BCCD, we need a new scoring method to estimate the reliability of causal relations.

There are several scoring methods in the literature for mixtures of discrete and continuous variables. Most of these methods either rely on strict assumptions about the structure of the network that do not apply in practice, such as forbidding structures in the network with a continuous variable as a parent having a discrete variable as child, or are time consuming and/or memory inefficient.

2 Methods

In this paper we propose a fast and memory efficient method to score DAGs with both discrete and continuous variables using BIC score, under the assumption that the relationships between these variables are monotonic. This appears to be a reasonable assumption for many real-world data sets. The BIC score can be decomposed into the sum of two components, the mutual information $I(X_i, Pa_i)$ with Pa_i the parents of node X_i and $Dim[\mathcal{G}]$ the number of parameters necessary to estimate the model. The first component measures the goodness of fit, and the second penalizes the complexity of the model. To estimate BIC for a mixture of discrete and continuous variables we need to estimate the mutual information and the complexity penalty. We propose to approximate the mutual information based on the formula for continuous variables drawn from a Gaussian distribution [2]:

$$I(X_i, Pa_i) = -\frac{1}{2} \log \frac{|R|}{| R_{Pa_i} |} \ , \tag{1}$$

[1] The full paper has been published in *Proceedings of the Seventh European Workshop on Probabilistic Graphical Models.*, 2014.

where R is a correlation matrix between all variables and R_{Pa_i} is a correlation matrix between the parents of variable X_i. Estimation of the complexity penalty for the model containing a mixture of variables is reduced to calculation of the parameters in the regression model.

In addition to (1) we propose substituting Pearson correlation with Spearman. We assume hereby explicitly that the variables obey a so-called non-paranormal distribution: a multivariate Gaussian distribution on latent variables, each of which is related to the observed variables through a monotonic transformation. As shown in [3], conditional independence tests for non-paranormal data based on Spearman correlations are more accurate than those based on Pearson correlations for non-Gaussian continuous data. Ignoring the discreteness of the discrete variables does introduce some bias in the approximation of the BIC score, however, this bias hardly affects the scoring of the network structures.

The most computationally expensive part of the proposed scoring method is the calculation of the correlation matrix. However, one can compute the full correlation matrix once beforehand, which can then be stored and used to efficiently construct the correlation matrices for any subset of variables. The proposed method is thus computationally and memory efficient.

3 Results

To test our algorithm on simulated data, we chose two widely used Bayesian networks: the Asia Network and the Waste Incinerator Network to test the algorithm for discrete variables and a mixture of discrete and continuous variables, respectively. We randomly generated data for four different sample sizes: 100, 500, 1000, and 1500 and repeated our experiments 20 times. Performance was measured by PAG accuracy measure, that evaluates how many edges were oriented correctly in the output PAG. We also estimated the correctness of the skeleton by calculating the amount of correct, missing, and spurious edges of the resulting graph. Simulation studies showed that the proposed scoring method appears to rely on a reasonable approximation of mutual information. The small bias introduced to it hardly influences the outcome of the BCCD algorithm. To test the BCCD algorithm on real-world data, we used the data set collected for ADHD-200 competition that contains a mixture of discrete and continuous variables. The resulting network inferred by the BCCD algorithm provided the causal relationships between different factors and symptoms. Several publication in medical journals have been found that confirm these relationships. Based on this results, we conclude that the BCCD algorithm with the proposed scoring method can accurately estimate the structure of the Bayesian network for both simulated and real-world data.

Acknowledgments

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 278948.

References

[1] T. Claassen and T. Heskes. A Bayesian approach to constraint based causal inference. In *Proceedings of the UAI Conference*, pages 207–216. AUAI Press, 2012.

[2] T. M. Cover and J. A. Thomas. *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)*. Wiley-Interscience, 2006.

[3] N. Harris and M. Drton. PC algorithm for nonparanormal graphical models. *Journal of Machine Learning Research*, 14:3365–3383, 2013.

Flexibility and Decoupling in Simple Temporal Networks[1]

Michel Wilson [a] Tomas Klos [b] Cees Witteveen [b] Bob Huisman [b,c]

[a] *West Consulting, Delft, The Netherlands*
[b] *Delft University of Technology, Delft, The Netherlands*
[c] *Fleet Services, Nedtrain Company, Utrecht, The Netherlands*

1 Introduction

Scheduling problems occur in a wide range of applications, such as manufacturing, transportation, routing, education, etc. In a scheduling problem, we are asked to assign values to a set of time variables under temporal constraints. Such an assignment σ is called a *schedule*. We focus on a simple yet powerful model for expressing temporal information called the Simple Temporal Network (STN) [1]. An STN $S = (T, C)$ consists of a set of time variables $T = \{t_0, \ldots, t_n\}$, and a set C of binary constraints on those variables of the form $t_j - t_i \leq c_{ij}$, which limit the difference between the values that can be assigned to certain pairs of variables. Several computational problems on STNs—such as deciding whether an assignment that satisfies all constraints exists, generating an arbitrary schedule, or computing the earliest and latest starting times for all events—are efficiently solvable (see [3] for a detailed overview and thorough analysis of such problems).

For various reasons, it is often undesirable to give just a single, fixed-time schedule. Such a schedule is vulnerable to disturbances that may occur during execution of the schedule. In such commonly occurring circumstances, computing some form of *flexible* schedule is desirable [4]. We propose the use of an *interval schedule*, which assigns to each variable not a single value, but rather an interval of allowed values. The sum of the lengths of all variables' intervals can then be taken as a measure of the *flexibility of an interval schedule*, and the maximum value of this sum over all interval schedules as a measure of the *flexibility inherent in the STN*. Naturally, we are interested in computing the flexibility of an STN, along with an interval schedule achieving it, but the question is which intervals we should assign.

2 A novel flexibility metric for STNs

The idea of an interval schedule is not new. It has been pointed out that if a scheduling formalism enjoys *decomposability*, then this property may be used to achieve some form of flexibility by assigning intervals to events [1, 4, 5]. Decomposability means that for every partial assignment there exists an extension to a full assignment that satisfies all constraints—this property holds for (minimal) STNs. Suppose that for every variable t in an STN S we compute the interval $[est(t), lst(t)]$ of allowed values between t's earliest and latest starting times. Then for every value we assign to some variable u from its interval, there exists an assignment to the other variables in S that satisfies all of S's temporal constraints. However, the sum of the lengths of these intervals will usually *overestimate* the flexibility of S, because not all values in all variables' intervals may be combined to form valid schedules: events have *dependencies* that a flexibility metric should take into account. To illustrate this, consider an STN with three variables t_1, t_2, and t_3, which all have to be scheduled between time points 0 and 5, while they also

[1] The full paper appeared in *Artificial Intelligence* (Volume 214, pages 26–44, 2014), as an invited extended version of our 2013 IJCAI paper "Flexibility and Decoupling in the Simple Temporal Problem," which won an *IJCAI Distinguished paper award*.

have to be scheduled in the order $t_1; t_2; t_3$. Now $est(t_i) = 0$ and $lst(t_i) = 5$ for all i, so a schedule that assigns these intervals would have a 'naive' flexibility $\mathit{flex}_N = 15$. But of course this flexibility metric disregards the ordering constraint, under which the total flexibility is more appropriately computed as $\mathit{flex} = 5$: the range from 0 to 5 should be *divided* among the three variables.

To remedy this, we propose a flexibility metric flex for STNs which is based on an assignment of intervals that *does* take dependencies between events into account. In our paper, we show that our metric flex is the unique metric that satisfies a set of intuitive rationality postulates, in particular one that says that *every combination* of values picked from several variables' intervals can be extended to a full schedule, which means flex takes into account event dependencies. Then we show how these intervals can actually be computed. If S contains a constraint of the form $t_j - t_i \leq c_{ij}$, then the intervals $[\ell_i, u_i]$ and $[\ell_j, u_j]$ assigned to t_i and t_j, respectively, should be such that $u_j - \ell_i \leq c_{ij}$. In that case, the constraint can not be violated, irrespective of which values from their respective intervals are assigned to t_i and t_j. To compute the intervals for an STN $S = (T, C)$, we consider all interval *boundaries* as variables to be assigned a value, as the solution of a *derived* STN $S' = (T', C')$. The set T' contains, for every $t \in T$, *two* variables t^- and t^+ (encoding the boundaries for t's interval), and the set C' contains the constraints mentioned above, that $t_j^+ - t_i^- \leq c_{ij}$ for every constraint $(t_j - t_i \leq c_{ij}) \in C$, plus a constraint $t^- - t^+ \leq 0$ for every $t \in T$, which ensures that lower bounds are not larger than upper bounds. Now, every assignment of values to the variables in T', constitutes an interval schedule for the variables in S. To find an interval schedule with maximal summed width, we simply solve the STN S' as a linear program with objective function $\sum_{t \in T}(t^+ - t^-)$.

3 Temporal decoupling without loss of flexibility

This approach is also useful in the context of *distributed scheduling*, where the set T is partitioned among k autonomous agents, each of whom is responsible for scheduling a non-empty, disjoint subset $T_i \subseteq T$. The agents are assumed to be capable of finding a local schedule that satisfies their *intra-agent constraints* (those involving only variables under their control), but it is not in general the case that the union of different agents' local schedules will also satisfy all *inter-agent constraints*, those involving variables under the control of two *different* agents. A solution that has been proposed for this problem in the context of STNs is *temporal decoupling* [2]. This method involves computing unary constraints on the variables involved in inter-agent constraints, such that these unary constraints together *imply* the inter-agent constraint, thus rendering it obsolete. A set of local STNs $\{S_i\}_{i=1}^k$ (containing original plus decoupling constraints) for which it holds that the union $\bigcup_{i=1}^k \sigma_i$ of *all possible* local schedules σ_i that satisfy the constraints in S_i, is a schedule for S, is called a *decoupling* of S.

One may expect that adding constraints to achieve a decoupling will lead to loss of flexibility: adding a constraint can not increase the range of allowed values for variables, and may actually decrease it. In our paper we show that once the flexibility of an STN S has been found, and a partitioning of T among k agents is given, a decoupling of S with maximal flexibility $\sum_{i=1}^k \mathit{flex}(S_i)$ can be found in $O(k)$ time. More importantly, and contrary to widespread belief in the community, this optimal decoupling need not suffer any loss of flexibility: We show how to find a decoupling where $\sum_{i=1}^k \mathit{flex}(S_i) = \mathit{flex}(S)$.

References

[1] R. Dechter, I. Meiri, and J. Pearl. Temporal constraint networks. *Artificial Intelligence*, 49, 1991.

[2] L. Hunsberger. Algorithms for a temporal decoupling problem in multi-agent planning. In *Proceedings AAAI*, 2002.

[3] L.R. Planken. *Algorithms for Simple Temporal Reasoning*. PhD thesis, TU Delft, 2013.

[4] N. Policella, S. Smith, A. Cesta, and A. Oddi. Generating robust schedules through temporal flexibility. In *Proceedings ICAPS*, 2004.

[5] M. E. Pollack and I. Tsamardinos. Efficiently dispatching plans encoded as simple temporal problems. In I. Vlahavas and D. Vrakas, editors, *Intelligent Techniques for Planning*. Idea Group, 2005.

Robot Mood is Contagious: Effects of Robot Body Language in the Imitation Game[1]

Junchao Xu[a] Joost Broekens[a] Koen Hindriks[a] Mark A. Neerincx[a,b]

[a] *Delft University of Technology, 2628CD Delft, the Netherlands*
[b] *TNO Human Factors, 3769DE Soesterberg, the Netherlands*

Abstract

We have designed a framework for bodily mood expression of a robot that integrates affective cues with functional behaviors without interrupting these behaviors. The framework is based on a parameterized behavior model in which the parameters control the poses and motion dynamics of a behavior. Modulating those parameters results in different behavior appearance through which robot mood is expressed. This study addresses the question whether robot mood displayed during an imitation game can (a) be recognized by people and (b) produce contagion effects. The gestures performed by the robot in the game were modulated to display either a positive or a negative mood. The task difficulty was also varied. The results show that participants are able to differentiate between positive and negative robot mood. Moreover, self-reported mood matches the mood of the robot in the easy task condition. Additional evidence for mood contagion is provided by the effect of robot mood on participants' task performance.

1 Introduction

Expression of internal states (e.g., affect, intention, or rationale) through body language is an important ability of social robots [1]. It has in particular been a challenge to express mood by body language at an arbitrary time during the execution of a task. To address this, we have designed behavioral cues and integrated them with functional behaviors needed to perform the task. Our approach is to "stylize" behaviors by modulating behavior parameters [2], [3] instead of using additional body movements, which have typically been used to express bodily emotion in research. Mood is a relatively long-lasting and stable affective state. Integrating mood into the body language of a robot provides the robot with not only an alternative but also a more stable channel for communicating affective information.

To figure out which behavior parameters have the potential to express mood and how to modulate these parameters to express a specific mood, we conducted a user study [2] in which participants were asked to modulate behaviors to match given valence levels by adjusting the parameters. The video clips of the resulting design can be found on our website[2]. The resulting settings were evaluated in a recognition experiment [4] in a laboratory setting without a context. The results showed that not only valence but also arousal can be well recognized.

This paper addresses the following research questions: 1) can people, while interacting with a robot, recognize mood from robot behaviors that are modulated to express positive or negative mood? 2) What are the effects of robot mood on someone who is interacting with that robot?

2 Affective Body Language in the Imitation Game

The mood expression was integrated with the gestures used in an imitation game [5], in which a human player imitates the gestures performed by the robot in a laboratory environment. A video clip of the

[1] The full paper [5] has been published in *Proceedings of the international conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, pages 973–980, 2014.
[2] http://ii.tudelft.nl/~junchao/moodexpression.html

gestures used in the experiment and gestures modulated by a continuously changed mood is available on our website[3]. Results not only confirmed that people can recognize robot mood from the body language in an interaction context, even when they were not primed to pay attention to the expression, but also shows evidence of a "mood contagion" effect [6]: participants' own mood matches with the mood of the robot in the easy task condition. Moreover, the robot mood had an effect on game performance: in the negative mood condition participants performed better on difficult tasks than in the positive mood condition. As a behavior measure, this result further supports the contagion effect.

3 Conclusion and Future Work

This study shows that it is feasible to use parameterized behavior to express a robot's mood in an HRI interaction scenario. Results show that people are clearly able to distinguish between positive and negative robot mood. Our results also suggest that mood contagion takes place between the robot and the human. The evidence for this contagion effect consists of: 1) participants self-reported mood matching that of the robot, and 2) participants' task performance being lower in the positive robot mood condition compared to the negative robot mood condition replicating a well-known mood-cognition effect.

Another interesting aspect that we found in our study is that participants attributed the robot mood to various factors that were not manipulated. In a more complex interaction scenario, even more factors could influence the results. An ongoing effort is to apply mood expression in a robotic tutor application, in which the robot NAO gives a lecture and interacts with audiences using quiz questions in a university classroom. The co-verbal gestures are modulated to express the robot mood. The robotic tutor scenario is close to real life and the robot mood is expressed through a broad range of modulated robot gestures over a longer period of time (30 min), and the robot interacts with a large group of individuals instead of a dyadic interaction. We have tested the mood expression of the robotic tutor in a positive and a negative mood condition [7], and observed the differences of participants' own valence and arousal between conditions, which suggests that the mood expression can be used to shape the interaction affectively. Moreover, the audiences' ratings of the lecturing quality and gesture quality of the robot are higher in the positive condition, suggesting body language of a robotic teacher is important for the rating of the robot's teaching quality. Additional materials including videos are available on our website[4].

References

[1] C. Breazeal, A. Takanishi, and T. Kobayashi, "Social Robots that Interact with People," *Springer Handb. Robot.*, pp. 1349–1369, 2008.

[2] J. Xu, J. Broekens, K. Hindriks, and M. A. Neerincx, "Mood expression through parameterized functional behavior of robots," in *RO-MAN, IEEE*, 2013, pp. 533–540.

[3] J. Xu, J. Broekens, K. Hindriks, and M. A. Neerincx, "The Relative Importance and Interrelations between Behavior Parameters for Robots' Mood Expression," in *ACII, Geneva*, 2013, pp. 558–563.

[4] J. Xu, J. Broekens, K. Hindriks, and M. A. Neerincx, "Bodily Mood Expression: Recognize Moods from Functional Behaviors of Humanoid Robots," in *ICSR, Bristol, UK*, 2013, pp. 511–520.

[5] J. Xu, J. Broekens, K. Hindriks, and M. A. Neerincx, "Robot Mood is Contagious: Effects of Robot Body Language in the Imitation Game," in *Autonomous Agents and Multi-Agent Systems (AAMAS)*, 2014, pp. 973–980.

[6] R. Neumann and F. Strack, "'Mood contagion': the automatic transfer of mood between persons.," *J. Pers. Soc. Psychol.*, vol. 79, no. 2, pp. 211–223, 2000.

[7] J. Xu, J. Broekens, K. Hindriks, and M. A. Neerincx, "Effects of Bodily Mood Expression of a Robotic Teacher on Students," in *IROS*, 2014.

[3] http://ii.tudelft.nl/SocioCognitiveRobotics/index.php/ImitatGameMood

[4] http://ii.tudelft.nl/SocioCognitiveRobotics/index.php/RoboTutorMood

Part III

Demos (C)

Using Facial Expressions for Personalised Gaming

Paris Mavromoustakos Blom Sander Bakkes Diederik M. Roijers

Informatics Institute, University of Amsterdam, the Netherlands

Artificial intelligence (AI) in games ideally provides satisfactory game experiences for all players, regardless of gender, age, capabilities, or experience [4]: it personalises games, i.e., the game experience is continuously tailored to fit the individual player. A major challenge in game personalisation is how to build and estimate a model of how game elements affect the player. In particular, we are interested in which elements have a positive effect on the *affective state* of the player. However, effectively estimating information about this affective state from the information that is available in online gameplay is a major challenge. In online gameplay only implicit feedback on the appropriateness of the personalisation actions is available, i.e., the AI can only observe the player interacting with the game, while not being provided with labels on the player's *affective state*, and explicitly asking the player is too intrustive for the player to fully enjoy the game.

In this demo, we show how game personalisation techniques can leverage novel computer vision-based techniques to *unobtrusively* infer player experiences automatically based on facial expression analysis. Specifically, we employ algorithms for tailoring the affective game experience to the individual user that (1) leverages the proven INSIGHT facial expression recognition SDK as a model of the users affective state[1], and (2) employ this model for guiding the online game personalisation process. Specifically, we use the emotion *anger* versus *happiness* (or *neutral*) to measure user experience, and adapt the game accordingly. These algorithms have been previously published in [2].

In this demo, participants can play the video game INFINITE MARIO BROS, while it is being personalised using an AI. The AI typically decreases challenge levels of certain game elements when the face of the user shows anger, and typically increases challenge levels of these game elements when the face of user is neutral or looks happy.

1 Approach

In this section, we describe the algorithmic approach that we use for the demonstration, as previously published in [2], which builds on the perspective that computer vision techniques can provide to automatically infer gameplay experience metrics. The first step in doing so is to recognize facial expressions. This is a well-studied domain in computer vision with techniques that can reach a high level of accuracy and robustness. E.g., Buenaposada et al. [3] have reported an 89% recognition accuracy on average in video sequences in unconstrained environments with strong changes in illumination and face locations.

In this demonstration, the challenge level of a game is balanced by adapting the *content* that is placed within the game environment dependent on facial expression analysis, i.e., we use procedural content generation for tailoring to the individual player [1, 2]. To this end, we use an AI to assess, *online* and *unobtrusively*, which adaptations are required for optimizing the individual players experience while the game is in progress. We use a typical video game: INFINITE MARIO BROS[2] — an open-source clone of the classic video game SUPER MARIO BROS. In particular, we use the 2011 Mario AI Championship game engine of INFINITE MARIO BROS, with two further enhancements [1]. First, it now is able to procedurally generate segments of Mario levels while the game is in progress, as illustrated in Figure 1. Second, we can now inject short chunks of specific game content: (1) a straight chunk, with enemies and jumpable blocks, (2) a hill chunk, also with enemies, (3) a chunk with tubes and enemy

[1] http://sightcorp.com/

[2] http://www.mojang.com/notch/mario

plants, (4) a jump, and (5) a chunk with cannons. Each type of chunk can have six distinct challenge levels, expressed as a parameter value in $[0, 5]$. The challenge level of the chunk increases monotonically with the parameter value. In online gameplay, the only action that the personalisation AI can take is to output a vector of five integers (chunk parameters on the interval $[0, 5]$) to the procedural process that generates the next level segment. While this action space is relatively small, its resulting expressiveness ranges from extremely easy to nearly impossible level segments.

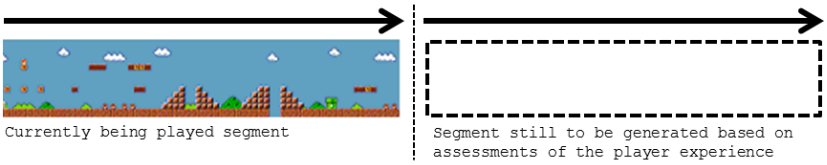


Figure 1: *Generating segments for INFINITE MARIO BROS.*

As input, the personalisation AI uses classifications of the human players facial expressions. In order to map these classifications to appropriate in-game challenge levels, we employ a Gradient Ascent Optimisation (GAO) technique [2], for optimising the challenge levels for each chunk type in the game, such that human interactions with the content yield affective stances that we consider desirable (i.e., happiness), while minimising affective stances that we do not consider desirable (i.e., neutrality and anger).

After a game segment has been completed by the human player, the probability-distribution vector of the measured emotional stances are retrieved for each individual chunk. The emotions taken into consideration for the present experiments are (1) neutrality, (2) happiness, and (3) anger. When a game segment is finished, the emotion vectors – of each individual chunk – of the recently played (finished) segment, plus the previously completed segment, are fed into the algorithm. For each emotion that is taken into consideration (neutrality, happiness and anger), the difference between its current and previous iteration value is obtained. Next, the maximum of these three differences is determined. Since the three emotions are equally weighted, the maximum value calculated could be considered as the "most significant" change in emotional status of the user between two game segments. We map this emotional change to an action of the AI, i.e., a change in challenge level for the chunk types of the segment that has just been completed by the player on the interval $[-5, 5]$. (For chunks that did not appear in the last segment, we do not change the challenge level.) In practice, the algorithm will increase the game challenge level if the probability estimate of an emotion is higher in timestep t compared to $t - 1$.

A pilot study [2] indicated that this approach decreases specific challenge levels in the face of user anger, and increases specific challenge levels in the face of user neutrality or happiness as expected, and appears stable in the face of classification noise. A significant majority of participants in the study prefered the personalised system over a non-adaptive system, except in cases when anger (frustration) of the human participants was not expressed in terms of facial expression, but in terms of hand gesturing, verbal actions, or head movements that prevented facial expressions from being assessed accurately.

2 Demo

In the demo, participants can play INFINITE MARIO BROS., while the personalisation AI adapts the game on the basis of the facial expressions of the participant. The demo requires a computer with a 1.6GHz dual core processor or better, 2GB RAM, a webcam with a resolution of at least 640×480, and an active internet connection. The lighting is controlled by applying a stable light source, and screens around the player in order to avoid sudden fluctiations that might happen due to passing spectators.

References

[1] Sander Bakkes, Shimon Whiteson, Guangliang Li, Efstathios Charitos George Viorel Visniuc, Norbert Heijne, and Arjen Swellengrebel. Challenge balancing for personalised game spaces. In *Proceedings of the 6th IEEE Consumer Electronics Society Games, Entertainment, Media Conference (IEEE-GEM 2014)*, 2014. To appear.

[2] Paris Mavromoustakos Blom, Sander Bakkes, Chek Tien Tan, Shimon Whiteson, Diederik M. Roijers, Roberto Valenti, and Theo Gevers. Towards personalised gaming via facial expression recognition. In *Proceedings of the 10th AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE'14)*, pages 30–36, 2014.

[3] José M. Buenaposada, Enrique Muñoz, and Luis Baumela. Recognising facial expressions in video sequences. *Pattern Analysis and Applications*, 11(1):101–116, October 2007.

[4] D. Charles, A. Kerr, M. McAlister, M. McNeil, J. Kücklich, M. Black, A. Moore, and K. Stringer. Player-centred game design: Adaptive digital games. In *DIGRA*, pages 285–298, 2005.

Teaching Mario.
Demonstrating the effectiveness of human guidance in reinforcement learning

Roland Meertens [ab]

[a] *Pinch of Intelligence*
www.pinchofintelligence.com
[b] *Artificial Intelligence*
Radboud University Nijmegen

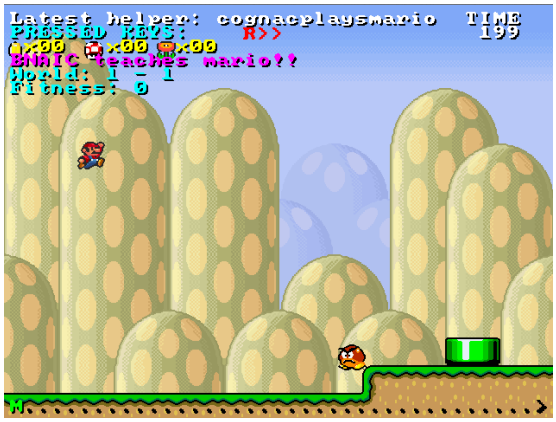
Abstract

Using human input has been shown to be very valuable for reinforcement learning. This demonstration proposal describes an implementation of an extension to the q-learning algorithm that learns a policy to play the classic Mario game. Visitors of the BNAIC (Belgium Netherlands Artificial Intelligence Conference) are able to download an application on their mobile phone that can give input to Mario. Using input of the participants, Mario will choose smarter actions and learn more efficiently.

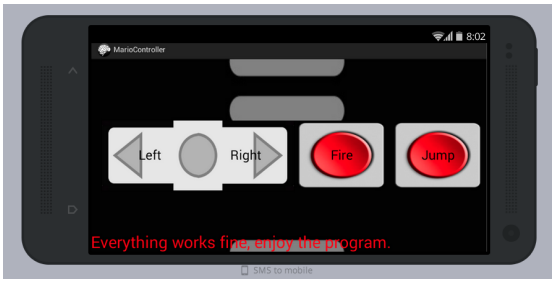
1 Introduction

The field of model-free reinforcement learning assumes that it is possible to create a mapping between the state of the world (represented using several features) and what action (from a fixed set of actions) to take. The rate in which the computer learns can be greatly increased [3] by using human guidance to select the action that the algorithm should explore. An experiment by Smart and Kaelbling [2] demonstrates this, their mobile robot was able to learn a navigation policy faster and with more accuracy when using human input than when randomly exploring. The reason that the computer learns faster is because it incorporates the human knowledge and reasoning about the task to allow for adaptive input.

For the BNAIC 2014 (Belgium Netherlands Artificial Intelligence Conference), we created a demonstration to show the effectiveness of human guidance, in which visitors are able to choose the actions the



(a) A screenshot of the modified Mario game



(b) The look of the application

algorithm explores. The demonstration uses the Mario AI framework (discussed later this section) and a phone application that all interested visitors have to download. This allows a great amount of people to give instructions to Mario at the moment they believe it is necessary to give instructions.

The Mario AI benchmark is a clone of the classic game Super Mario Bros. The gameplay consists of moving Mario through the level while jumping (and possibly throwing fireballs). The main goal of the game is to reach the end of the level while gathering as much points as possible. The game asks the player to give instructions to Mario (walk left, walk right, jump, fire) in each frame, and displays the result of the action to the user each resulting frame.

The Mario AI benchmark includes an evaluation function to test the performance of the machine learning algorithm. During the evaluation round the algorithm will be tested to see how well the algorithm performs on new levels (to prevent the algorithm overfitting one level). The evaluation function allows us to to see how the algorithm performs and how fast the algorithm learns.

Many solutions to the benchmark can be found online. One of the open source solutions that can be found online is the q-learning algorithm created by Stanford students participating in a machine learning course (CS229) [1]. We extended their algorithm by adding the possibility to let a human override the decision of the agent. To do this users install an application on their mobile phone(see image 1b). This application slightly resembles the controller that was used to control the first game of Mario and contains the buttons for Mario's possible actions. The presses of the buttons are sent to the application and override the decisions of the agent.

2 Method

To enable all visitors to the conference to participate in the demonstration a screen and computer will be placed in the central hall of the conference building. On this screen the game will be visible to all by-standing participants. At the start of the conference the goal of the game will be explained, and instructions are given about how to download the application.

Mario executes a normal q-learning [4] algorithm, randomly selecting an action 'some' percent of the time (according to the exploration chance parameter). Participants can press buttons on their application and hereby send actions to the game after which the Mario agent will execute this action, hereby overriding the normal q-learning exploration algorithm. If Mario received multiple actions he will select the action that is most chosen and will put one of the names of the participants who gave this action on the screen. Each time Mario dies a short evaluation run will be performed and the resulting fitness will be shown on the screen.

References

[1] Yizheng Liao, Kun Yi, and Zhe Yang. CS229 Final Report Reinforcement Learning to Play Mario. Technical report, 2012.

[2] W.D. Smart and L. Pack Kaelbling. Effective reinforcement learning for mobile robots. *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No.02CH37292)*, 4(May):3404–3410, 2002.

[3] AL Thomaz and Cynthia Breazeal. Reinforcement learning with human teachers: Evidence of feedback and guidance with implications for learning performance. *AAAI*, pages 1000–1005, 2006.

[4] Christopher J. C. H. Watkins and Peter Dayan. Q-learning. *Machine Learning*, 8(3-4):279–292, May 1992.

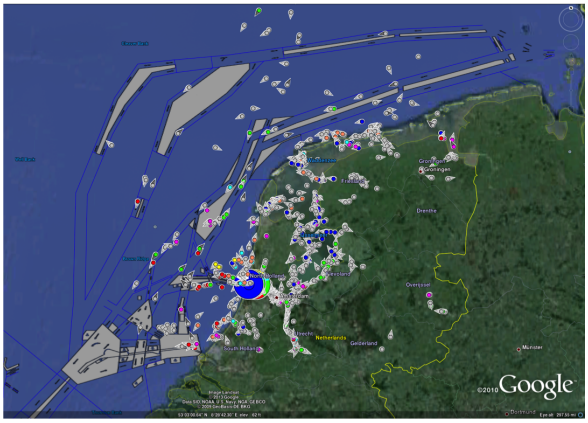
Enhancing operational work in maritime safety-and-security tasks

Steffen Michels [a] Marina Velikova [b] Bas Huijbrechts [b] Peter Novák [c]
Jesper Hoeksma [d] Roeland Scheepens [e] Jan Laarhuis [f]
André Bonhof [f]

[a] *Radboud University Nijmegen, The Netherlands*
[b] *Embedded Systems Innovation by TNO, The Netherlands*
[c] *Delft University of Technology, The Netherlands*
[d] *Free University Amsterdam, The Netherlands*
[e] *Eindhoven University of Technology, The Netherlands*
[f] *Thales Nederland B.V., The Netherlands*

1 Problem description and aims

On a yearly basis, at the North Sea there are thousands of ship movements, to and from Dutch ports, as well as vessels crossing over towards ports in other countries via shipping lanes that are considered the most busiest in the world (see the picture on the right). Adding complexity, the same shipping lanes are located near extreme sensitive environmental areas such as the Dutch Wadden Sea. Maritime safety and security demands understanding of all relevant activities occurring at sea where, compared to land, marked delineations, physical barriers and associated fixed sensors often lack or are only virtually present.



Such understanding involves the continuous monitoring of vessels to detect and predict certain events of interest such as accidents (e.g., collision) or illegal activities (e.g., smuggling), whose consequences can be devastating in terms of human life losses, financial costs or damage to the environment. This requires the real-time collection and processing of a vast amount of information, which is a great challenge for human operators.

Maritime safety-and-security systems have emerged to facilitate timely human decision-making. While recent systems allow automatic identification and warning of abnormalities, they lack capabilities for real-time prioritisation of the application tasks, selection and alignment of relevant information, and efficient reasoning at a situation level.

Such intelligent capabilities are embedded in the innovative system we developed within the Metis project based on the employment and integration of state-of-the-art artificial intelligence approaches. The general architecture of the Metis system is described in details in [1].

2 Deployed technology

The overall intelligence of the Metis system lies in the unique integration of the following components:

(1) Data collection & **semantic alignment**. Data about a vessel under investigation from heterogeneous and (un)structured sources is being automatically retrieved by the system. These typically involve messages broadcasted by the vessel itself, radars, ship tracking websites, and external vessel databases. We deploy *natural language processing* techniques to detect valuable clues inside large bodies of unstructured documents, such as newspapers or intelligence reports, that might indicate or confirm suspicious behaviour in the past. A detailed description of this component is described in [2].

(2) Interpretation of noisy, incomplete and dynamic maritime-related data. The goal is to best possibly predict (i) the true values of the ship's attributes such as identification number, name and type, and (ii) the ship's behaviour in terms of threats such as collision and smuggling, which are used to warn the operator. Given the heterogeneity and uncertainty in the collected ship information, and the dynamic number of sources and ships observed, we chose *first-order probabilistic logic* as a modelling technique. An earlier version of this component with tests of its performance is presented in [3].

(3) (Re)-configuration of the work of the system's components. The Metis system relies on a number of heterogeneous external information sources that can fail, become inaccessible over time, and their access can incur financial costs (e.g., pay-per-request). The goal of the re-configuration component is to compute relevant and resource-saving query plans depending on mission goals and costs to provide cost-effective data collection and interpretation. To do so, the reconfigurator models the Metis structure as a *non-monotonic multi-context system*. A formal description of the underlying theoretical principles of cost-aware reconfiguration is given in [4].

To allow autonomous and continuous running of the Metis system, the communication between its components is provided by a data distribution infrastructure. Finally, the interaction with the operator is supported by an advanced user interface that displays a tailored operational picture of the current situation. The results of the projects are validated into a demonstrable proof of concept and deployed as a plugin in the Thales' command-and-control industrial platform Tacticos.

3 The demonstrator

The full operation of the Metis system will be demonstrated with real-world vessel data, starting from automated data collection and interpretation to reconfiguration and advanced visualisation of the results obtained. The data is obtained from ship identification messages, a world register of ships, ship tracking websites such as marinetraffic.com and PressAssociation news about maritime events. A number of realistic scenarios will illustrate the complexity of the domain and the added value of the Metis system.

Acknowledgements

This work was supported by the Dutch national program COMMIT. The research was carried out as a part of the Metis project under the responsibility of the TNO-Embedded Systems Innovation, with Thales Nederland B.V. as the carrying industrial partner. We thank Tom Regelink for his valuable contribution on the system integration.

References

[1] T. Hendriks and P. van de Laar. Metis: Dependable cooperative systems for public safety. *Procedia Computer Science*, 16:542–551, 2013.

[2] J. Hoeksema and W.R. van Hage. A case study on automated risk assessment of ships using newspaper-based event extraction. In *Workshop of DeRiVE*, volume 1123, pages 42–52, 2013.

[3] S. Michels, M. Velikova, A. Hommersom, and P. Lucas. A decision support model for uncertainty reasoning in safety and security tasks. In *IEEE Intern. Conference on SMC*, pages 663–668, 2013.

[4] P. Novák and C. Witteveen. On the cost-complexity of multi-context systems. *CoRR*, arXiv:1405.7295, May 2014. `http://arxiv.org/abs/1405.7295`.

An Implementation for Distances between Labellings in Abstract Argumentation

Mikolaj Podlaszewski [a] Yining Wu [a]

[a] *University of Luxembourg, Luxembourg*

Abstract

With this demonstrator, we present an implementation of formal argumentation theory that not only is able to measure the distances between any two complete labellings and the distances between an arbitrary complete labelling and a multiset of complete labellings, but is also able to generate all complete labellings and argument issues of an argumentation framework.

1 Introduction

In an argumentation framework, more than one set of arguments can be selected with respect to complete semantics. Essentially, a complete labelling can be seen as a subjective but reasonable position that an agent can take with respect to which arguments are accepted, rejected or undecided. The set of all complete labellings thus stands for all such positions. Different individuals may provide different evaluations regarding what should be accepted and rejected. Therefore, different individuals may take different positions. The distance between two complete labellings expresses how different two reasonable positions are. Such distances make it possible to choose a reasonable position that is closest to the collectively rational decision. Therefore it can serve as a way for judgment aggregation.

In order to assess the difference between complete labellings, many distance measures have been proposed [1, 2]. However, it is not an easy task to manually calculate the distances even for a small argumentation framework. A software is missing to automate the calculation. Ideally, it should be able to support different distance measures and at the same time provide a user friendly output. In this way, people can quickly grasp the difference between complete labellings from various perspectives and thus make a good decision. In order to provide such a software, we developed the current demonstrator.

2 Background Theory

The current demonstrator implements complete semantics, issue semantics [1], betweenness distances [1] and sum-based distance based on the argument-wise label difference, as well as based on the issue-wise label difference. It also supports four MO aggregation methods [3] that minimise the distance to ensure a rational collective outcome.

Given an argumentation framework $AF = (Ar, def)$, a complete labelling is a function that assigns to each argument exactly one label $(\texttt{in}, \texttt{out}, \texttt{undec})$ such that for each argument, the argument is labelled `in` iff all its defeaters are labelled `out`, and the argument is labelled `out` iff it has at least one defeater that is labelled `in`. An *issue* I of an argumentation framework AF is a subset of Ar such that for complete labellings: *(i)* all arguments in I are labelled `undec`, *(ii)* all arguments in I are labelled `in` or `out`, or *(iii)* the inverse labelling to *(ii)* occurs, in which those arguments labelled `in` become `out` and those labelled `out` are now `in`. The demonstrator can generate all complete labellings and issues which are subsequently used as the input of other functions, such as *sum*, *issue*, *betweenness* and MO aggregation functions. In Figure 1, we show the information flow of our demonstrator. Specifically, the output of the module on the left of an arrow is the input of the one on the right.

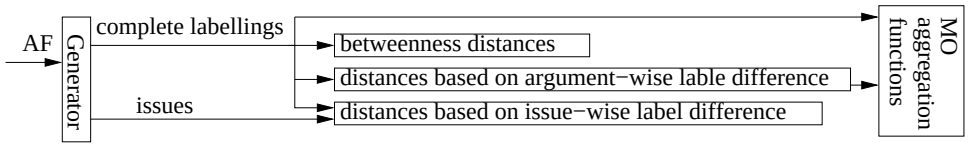


Figure 1: Architecture of the demonstrator

Given complete labellings and issues, the demonstrator is able to calculate the sum-based distances [1] between any two complete labellings. We assume we have some measure of disagreement *diff*$(\mathtt{x}, \mathtt{y})$ for any $\mathtt{x}, \mathtt{y} \in \{\mathtt{in}, \mathtt{out}, \mathtt{undec}\}$ between different labels, and then obtain the distance between two labellings by summing the differences between all arguments in the argumentation framework. In the current demonstrator, we use Hamming distance as the *diff*-measure. For distance via issue-wise label difference, only one argument is considered in each issue. We take one argument from each issue. The sum of the differences between two complete labellings of these picked arguments is issue-based distances of these complete labellings.

Except sum-based distance, the demonstrator can calculate distances based on the idea of "betweenness" [1]. The betweenness distances can be expressed by a graph in which the nodes represent complete labellings. Two complete labellings are connected with an edge if there is no complete labelling between them. A complete labelling L is between complete labellings L_1 and L_2 if L_1, L_2 and L are distinct and for each argument A, $L(A) = L_1(A)$ or $L(A) = L_2(A)$, or $L(A) = \mathtt{undec} \wedge L_1(A) \neq L_2(A)$. The betweenness distance between two complete labellings is the length of the shortest path between them.

In [2], the above distance methods for argumentation are employed to address problems of aggregation in argumentation by adapting the framework of Miller and Osherson [3] from binary judgment aggregation to several operators for aggregating argument labellings. Four MO aggregation methods [2], $\mathtt{P}, \mathtt{E}, \mathtt{F}$ and $\mathtt{O}$ that minimise distance to ensure the collective outcome is rational are implemented by the current demonstrator.

3 The Demonstrator

The demonstrator implements a library of functions:
- *sum*: returns the distances between two complete labellings based on the argument-wise label difference.
- *issue*: returns the distances between any two complete labellings based on the issue-wise label difference.
- *betweenness*: returns the betweenness distance of any two complete labellings and a visualized graph for betweenness distance.
- MO aggregation functions: each returns a set of labellings that are closest to the collective outcome.

These functions can be easily employed in other applications which quantify disagreement in argument-based reasoning and studies about the aggregation in abstract argumentation. As an additional piece of functionality, the demonstrator is able to generate all complete labellings and issues.

Acknowledgement: Y. Wu is supported by FNR Luxembourg, project "Socio-Technical Analysis of Security and Trust", C11/IS/1183245, STAST.

References

[1] R. Booth, M. Caminada, M. Podlaszewski, and I. Rahwan. Quantifying disagreement in argument-based reasoning. In *Proc. AAMAS*, 2012.

[2] R. Booth and M. Podlaszewski. Using distances for aggregation in abstract argumentation. In *Proc. BNAIC*, 2014.

[3] M. Miller and D. Osherson. Methods for distance-based judgment aggregation. *Social Choice and Welfare*, 32(4):575–601, 2009.

BNAIC
Interpreting eeg signals using Artificial Intelligence

Felipe Gomez [a] Ann Nowe [b] Yann-Michael De Hauwere [c]

Peter Vrancx [d]

[a] *Vrije Universiteit Brussel, Pleinlaan 2, 1050 Elsene*

Abstract

The aim of the research was to implement AI techniques to portable devices that measure electroencephalogram (EEG) signals so as to classify a number of body movements. We applied a clue-based paradigm to record motor imaginary and actual movements of both the left and right foot as well as the left and right hand using an Emotiv EEG device. Extensive analysis showed that the current feature selection techniques used in portable devices fail to characterise the signals for learning due to insufficient pre-processing. However, our research demonstrated that AI algorithms can perform both feature selection and movement prediction with little pre-processing of the EEG signals. A Neural Network(NN) and a Logistic regression(LGR) algorithm were trained and they obtained a classification accuracy of 95 percent using a novel feature selection technique. Based on the experiments, one could acknowledge that AI algorithms have the potential to create new signal representations that dramatically increase the accuracy of the predictions in both classifiers.

1 Demo

In the demo we will demonstrate how brain signals are interpreted by an AI system. The system implemented in the master thesis *Interpreting EEG Signals Using Artificial Intelligence* (Gomez, 2013) predicts whether the subject is thinking to move the left or right foot as well as the left or right hand. However, for this demo we will modify this implementation and map thoughts or movements such as up, down, left and right to the flight manoeuvres of a Quadcopter. The system developed in (Gomez, 2013) is composed of two subsystems: The training system which is in charge of recording the movements by stimulating the subject's brain with pictures, and the interpreter system which is in charge of interpreting the subject's thoughts. This system is further explained in section 2 and will be used to train the AI system that will be presented in this demo. In order to fully capture the essence of the experiment we will first provide an explanation and a demonstration of how the AI system was trained. This will include a 2 to 3 minutes online training. Second, we will use the trained AI system to send flight commands to a Quadcopter such as elevate, move left, move right, and move forward. The Quadcopter movements will be fixed to a discrete turning angle in order not to lose control over the Quadcopter. In addition, the Quadcopter will be in a static state (no movement) when the user thinks stop.

In order for the AI system to work it must be run on a device with Ubuntu_X86 12.04 installed. In addition, an Emotiv EEG headset research edition is needed. The system has not been tested with other Emotiv headset editions. This demo is expected to last approximately 10 minutes. A similar demo was also presented in the TV program *Het Lichaam van Coppens* on VTM in October 2014.

2 Experiments and Results

The experiments in (Gomez, 2013) were performed using the 4th BCI competition (Leeb et al., 2008) dataset, and the author's brain signals recorded by the EEG emotiv headset. The 4th BCI competition dataset contains the EEG signal recordings from 9 subjects. Each subject has 5 different data-sets each from a different session. Three sessions include the time positions of the labels and therefore they are used for training the classifiers. The other two data-sets are used for testing purposes. In addition, the experimental set-up followed a cue-based screening approach where only motor imagery (MI) of left hand and right hand movements were recorded. The second experiment was done by recording the first author's brain activity using an EEG Emotiv recorder (Emotiv). Similarly to the BCI experiments, this experiment followed a cue-based screening paradigm of the left hand (LH), right hand (RH), left foot(LF) and right foot(RF) movements of both motor imagery(MI) and motor movements(MM). The cue-screening was designed as follows. A black cross was shown at the centre of the screen. This was done to keep the subject's attention focused on an object without thinking of any movements. When the recording starts, a short beep warning sounds after 8 seconds announcing that a movement's cue is going to appear on the screen. At the 10th second a cue with a picture of a brain appears, this means that the subject has to imagine the movements of the cue. At the 14th second the brain on the cue disappears. At this moment the subject must start executing the same movement that he imagined during the MI. At the 18th second, the cue disappears leaving only the centred cross on the screen. At this moment the subject must focus back on the cross and try no to think or do any movement. This flow is shown in figure 1.

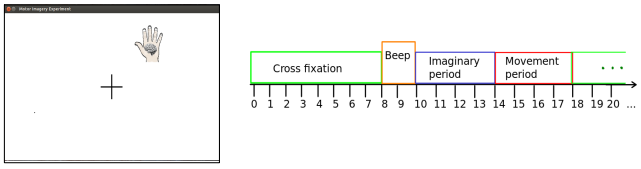


Figure 1: Emotiv recordings epochs

In this experiment we trained a logistic regression (Mohri et al., 2012) and a Neural Network (Schalkoff, 1997) to predict the motor imagery and actual movement of a subject. The training data was created by combining the outputs of the pre-processing and feature extraction steps. In order to measure the right accuracy performance of both algorithms a 10-fold cross validation was applied to the training set. Finally, for each test set we computed the fscore, accuracy and kappa values.

In the experimental results of the BCI dataset, we observed that using the spectrum representation the NN obtained a kappa of 0.34 and the LGR a kappa of 0.17. In the PSD representation we obtained a kappa of 0.43 in the NN and a kappa of 0.24 in the LGR. After extensive research on how neurons encoded data, we believe that the current feature selection techniques discard the way in which neurons process information. Research implies that information is spatially and temporally encoded when a stimulation such as right hand movement is available. For instance, when the brain is stimulated by a screen picture, the potentials that belong to this stimulation will fire at different times and probably with different frequencies as they are propagated to different regions of the brain. However, other types of stimulation such as heart signals are also being fired at the same time as the former stimulation. Therefore, when these signals arrive at the electrodes of the device they will be superposed. The spectral view of the signals let us know which frequencies are the most abundant when this stimulation was given. However, as there is always interference in the signals, the frequencies will always be slightly different at each training sample. Due to noise in the signal and the current pre-processing steps, the slightest change in these signal patterns becomes big making it difficult to classify.

Due to this disadvantage, we have created a novel feature extraction technique that exploits the spatial encoding of the neurons by creating a potential/time distribution of the experimental movements. This way we can model the propagation of the Action Potentials(AP) through the brain and dismiss other signals propagation such as lungs, heart, etc.. This novel feature extraction technique was performed directly on the raw signal. Using this new representation in the BCI dataset we obtained a kappa of **0.76** in the NN and a kappa of **0.72** accuracy in the LGR. Additionally, the previous mentioned techniques for signal processing failed to classify the movements in the EEG emotiv dataset whereas the new representation managed to obtain a kappa of **0.8** in the NN and a kappa of **0.74** in the LGR.

References

Emotiv. EEG specifications. `http://www.emotiv.com/eeg/download_specs.php`. URL `http://www.emotiv.com/eeg/download_specs.php`. [Online; accessed 1-July-2014].

Felipe Gomez. Interpreting eeg signals using artificial intelligence. Master's thesis, VUB, July 2013.

R Leeb, C Brunner, GR Müller-Putz, A Schlögl, and G Pfurtscheller. Bci competition 2008–graz data set b. *Graz University of Technology, Austria*, 2008.

Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of Machine Learning*. The MIT Press, August 2012. ISBN 026201825X.

Robert J. Schalkoff. *Artificial neural networks*. McGraw-Hill, 1997. ISBN 9780070571181.

Part IV

Thesis abstracts (D)

The Detection of Facial Expressions for Action Coordination[1]

T.C. Beinema [a] R. Dotsch [b] F.A. Grootjen [a]

[a] *Artificial Intelligence, Radboud University Nijmegen, The Netherlands*
[b] *Behavioral Science Institute, Radboud University Nijmegen, The Netherlands*

1 Introduction and Background

Facial expressions and gestures play an important role in non-verbal communication. In two-way interactions, facial expressions are typically used to convey the emotional state of an individual to an observer, while gestures can be used to coordinate actions between individuals. In the present study we investigate whether facial expressions might also be used for action coordination. If this is indeed the case and specific expressions can be identified in action coordination, there are various fields of research where they might be put to use, one of which is the field of affective computing.

Studies that involve (the use of) facial expressions commonly adopt the seven universal expressions of emotion that Ekman and Friesen distinguished. The expressions of anger, disgust, fear, happiness, sadness, surprise and contempt were consistently found in numerous studies on emotions across cultures[4] and are more generally considered a standard. Therefore, in the present study, we decided to use these seven expressions together with the neutral expression.

An experiment was conducted which involved a coordination game[1], known as the Battle of the Sexes. In this game, each of the two players can choose option A or option B. If they make the same choice, they get points; if they do not make the same choice, they do not get points. If they both choose A, player 1 will get 3 points while player 2 gets 2 points. If they both choose B, it is the other way around. This type of game requires participants to interact because they only get points if they both choose the same option.

Previous research on the effect of verbal communication in the Battle of the Sexes is described for example in Cooper et al.[2]. They show that communication has a positive effect on the process of the game in the form of increased scores. If the possibility for verbal communication does not exist, non-verbal communication might still provide a means for action coordination. In our experiment we adopted the following hypothesis: 'If there is non-verbal communication through facial expressions there should be an effect on the progress of the game in the form of an increased amount of hits.' As to what facial expressions are used for action coordination, it is not obvious what expressions can be expected. One of the main reasons is that facial expressions can have many functions, as is apparent from, for example, the existence of felt, false and miserable smiles[3].

In the present study, it was assumed that facial expressions could also have a different meaning depending on the moment they occur, i.e. before an action and after an action. We hypothesized that (1) prior to an event facial expressions are used to gain an opponent's trust or mislead the opponent, while (2) following an event facial expressions are used to convey feelings about the outcome.'We decided to focus more specifically on action coordinating facial expressions occurring as an initial reaction after an event. These initial reactions are presented in a limited amount of time after the event, which made it easier to select the correct timeframe for the data.

[1] The full bachelor thesis can been found online at the website of the Artificial Intelligence department of the Radboud University (http://www.ru.nl/ai).

2 Experimental Design

In the experiment 18 subjects took part. As for the environment and the experimental setup, there were a number of requirements. Thus the main requirement for the environment was that subject should not be able to communicate directly. It was therefore decided that the experiment was to take place in two adjacent rooms in a quiet location and the walls should provide a uniform background (so as to make the detection of subjects' faces in the camera recordings easier).

The custom software consisted of a playable version of the Battle of the Sexes game. Both subjects saw a panel with the game and a panel with camera images of the other subject's face on their screen. The software was run on a single laptop with build in webcam, which was put in one of the rooms. This was done to keep the measurements of all time stamps equal. In the other room a screen, a keyboard and an external webcam were placed, which were connected to the laptop. During the experiment, subjects played 97 rounds of the Battle of the Sexes in which each round consisted of both subjects making a choice. Subjects were not told how many rounds there were and they were told to get as high a score as possible, which required them to work together since they could only get points if they chose the same option. During the experiment, information about what happened in the game was saved with the corresponding timestamps. The same was done for the camera images.

The data were analysed using FaceReader[5]. For each frame a rating was obtained of how much each emotion was present. The ratings for the frames of interest were selected, i.e. the frame at the moment(t) the score for a round was displayed, and the frames capturing their initial reaction (t+200 ms, t+400 ms, and t+600 ms). We encoded eight patterns for each player for each round for the choices that were made in that round and three states (currently winning/losing/tied).

3 Results and Discussion

The results indicate that there was indeed non-verbal interaction. The experimental setup excluded almost all forms of interaction except through facial expressions, although subjects might have fallen into a pattern for the choices that they made. Since in the recorded videos subjects could been seen to, for example, smile in reaction to opponents' smiles, we may assume that communication did indeed take place through facial expressions, although the choice patterns probably were present as well. The statistical tests that were performed showed no significant change in facial expressions after the scores were shown in the game. This would suggest that the facial expressions did not change as an initial reaction to the display of the scores, but this remains to be verified in future research. We suggest that some adjustments be made to improve the experiment and the quality of the data gathered from it. The most important of them are to introduce a control group and to control the number of occurrences for choice patterns. Since subjects could be observed reacting to each other in the camera recordings, this would indicate that the data gathered was too complex to analyse using the methods that were applied.

References

[1] R. Cooper. *Coordination games*. Cambridge University Press, pages viii-x, 1999. (ISBN: 9780521578967).

[2] R. Cooper, D. V. DeJong, R. Forsythe, and T. W. Ross. Communication in the battle of the sexes game: some experimental results. *The RAND Journal of Economics*, pages 568–587, 1989.

[3] P. Ekman and W. V. Friesen. Felt, false, and miserable smiles. *Journal of nonverbal behavior*, 6(4):238–252, 1982.

[4] P. Ekman and W.V. Friesen. Constants across cultures in the face and emotion. *Journal of personality and social psychology*, 17(2):124–9, February 1971.

[5] B. Zaman and T. Shrimpton-Smith. The FaceReader: Measuring instant fun of use. In *Proceedings of the 4th Nordic conference on Human-computer interaction: changing roles*, pages 457–460. ACM, 2006.

Modeling and forecasting elections using topic models

Bas van Berkel

Radboud Universiteit Nijmegen, P.O.Box 9102, 6500 HC Nijmegen

Abstract

After elections, people want to know who will win the election as quickly as possible. During election night forecasts are made, based on both polls and the results from early polling stations. In this research a forecasting model based on pLSA is proposed. The model's forecasting performance is compared to a linear model's performance using the Dutch House of Representatives election. The model is also used to visualize and analyze voting profiles. The proposed model outperforms the linear model with a lower mean absolute error if 10% or more of the polling stations are observed. For 2.5% or less observed polling stations the linear model has a lower mean absolute error. The proposed model is also able to give insight into voting behavior by visualizing voter profiles. Thus, the proposed model is useful for both forecasting and modeling elections.

1 Introduction

These days polls form a big part of elections. During the counting of the results preliminary results are calculated. After the election voting behavior is analyzed, this is a prominent part of the news on elections in papers. This indicates an interest of the general public to know, what the election result will be as early as possible and what the underlying voting behavior is. During the early hours of the election night it is hard to predict an accurate outcome [1]. This is due the fact that different regions and types of municipalities have a different voting behavior. This issue is increased because the order in which results are declared does not necessarily result in a representative sample. In this research a topic model based on probabilistic Latent Semantic Analysis (pLSA) [2] is proposed to model voter profiles and to forecast elections. PLSA is used because it is possible to apply pLSA to higher order data. Topic models can be used to categorize documents into topics. When a word is in a given document, this is used as evidence for the document covering the topics which this word is part of. Although topic models are designed for, and mainly used to, analyze large sets of text documents, it is also possible to use them for other applications. To use pLSA for election modeling a translation is made from text categorization to election modeling. The main questions in this thesis are: can the proposed model give insight in voting behavior and can the proposed model be used to forecast elections, given partial results of those elections?

2 Methods and model

In the proposed model individual votes correspond to word occurrences in pLSA, a party corresponds to a word in the vocabulary, a polling station to a document, and voting profiles to the latent topics. pLSA is extended in the sense that there is a topic distribution over parties per election. The distribution over voting profiles per polling station is fixed over elections. The optimal values of both distributions are extrapolated using expectation maximization. We define $P(w_k|s_j, e_i)$ as the probability of a vote on party *k* in polling station *j* during election *i*:

$$P(w_k|s_j, e_i) = \sum_{m=1}^{V} P(w_k|v_m, e_i)P(v_m|s_j) = \sum_{m=1}^{V} \beta_{kmi}\theta_{mj}.$$

This is obtained by multiplying, for each profile, the probability that a voter in voting profile *m* votes on party *k* in election *i*: β_{kmi}, with the probability that a voter in polling station *j* is part of voting profile *m*: θ_{mj}. To forecast an election *i* the distribution over topics per polling station θ_{mj} is based on previous elections. The distribution over parties has to be estimated based on the observed polling stations s_j. To test the forecasting performance of the proposed model a linear model proposed by Pavía-Miralles [3] is used for comparison. Both models are tested with increasing numbers of municipalities (polling stations in the model description) with observed results: 2 (0.5%), 4 (1%), 10 (2.5%), 42 (10%), and 104 (25%). Both models are compared on their mean absolute error over 500 runs. For the proposed model 50 voting profiles are used, a smaller number causes underfitting, a larger number only slightly improves performance while increasing run time. The 2012 Dutch House of Representatives election is forecast with different numbers of previous elections used for fitting the distribution over voting profiles per municipality: 1, 3, 5, 8, 11, and 14. The models are compared on their mean absolute error over parties between the actual and forecasted results on a national level.

3 Results

With 2, 4, and 10 observed municipalities the linear model outperforms the proposed model with mean absolute errors of 0.0076, 0.0037, and 0.0023 compared to 0.0166, 0.0090 and 0.0035 for the proposed model. All with $p < 0.001$ ($T > 3.4$, $df = 499$). With 42 and 104 observed municipalities the proposed model outperforms the linear model with mean absolute errors of 0.0011, and 0.0006 compared to 0.0013, and 0.0010 for the linear model. Both with $p < 0.001$ ($T < -3.4$, $df = 499$). For all numbers of observed municipalities the performance of the proposed model improves with a decreasing number of previous elections. The proposed model can also be used to visualize voting behavior. The model forecasts elections using latent topic distributions. By visualizing these topics it is possible to observe shifts in voting behavior. When looking at the voting profiles over different election a clear continuation of parties in topics can be observed.

4 Conclusions

The proposed model can be used to forecast elections, it is even able to outperform existing models. With 0.5-2.5% of the municipalities observed, the linear model outperformed the proposed model, however, with 10-25% of the municipalities observed the proposed model outperformed the linear model. The proposed model can also be used to give insight into voting behavior by visualizing the probability distributions.

5 Acknowledgments

I would like to thank my supervisor, Prof. Tom Heskes, for his advice and valuable suggestions during my thesis project. I would also like to thank Dr. Louis Vuurpijl and Prof. Rob Eisinga for their help and suggestions on my thesis.

References

[1] Philip Brown and Clive Payne. Election night forecasting. *Journal of the Royal Statistical Society. Series A*, 138(4):463–498, 1975.

[2] Thomas Hofmann. Probabilistic latent semantic analysis. In *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence*, pages 289–296, 1999.

[3] Jose M. Pavía-Miralles. Forecasts from nonrandom samples: The election night case. *Journal of the American Statistical Association*, 100(472):1113–1122, 2005.

Using Neighbourhood-based Collaborative Filtering to Predict E-Learning Exercise Difficulty

Floris Devriendt [a] Ruben Lagatie [b] Maarten Devillé [a] Peter Vrancx [a]
Ann Nowé [a]

[a] *Vrije Universiteit, Pleinlaan 2 1050 Brussels, Belgium*
[b] *Televic Education, Leo Bekaertlaan 1, B-8870 Izegem, Belgium*

1 Research Question

Recommender Systems (RS) have found their way into many domains such as the recommendation of movies and music, or the recommendation of products in e-commerce websites. Over the last decade they have also touched upon the world of *Technology Enhanced Learning* (TEL) by recommending interesting learning materials such as novel books or informative articles [2]. Little research has been done to investigate the collaborative power of RSs as a classifier that can assist students in their online learning experience. In this dissertation we argue the use of a *Neighbourhood-based Collaborative Filtering* approach to determine whether a particular student can pass a particular exercise. Whereas RSs in education are usually used to guide users towards new content, the work in this dissertation will use RSs as a classifier on exercise difficulty. The main research question of the thesis is to investigate the impact of RSs in the educational world when regarding the use of RSs as tools to asses students' ability to solve unseen exercises.

2 Approach

A dataset was provided by an industry partner specialising in TEL systems. The company has developed several applications and tools to enhance the usage of technology in education. The dataset with real-life data originates from an educational platform for students and teachers with data from over the last five years. In particular, the data contains French language exercises targeted to users learning French as a second language. The dichotomous data has limited meta-data, but allows us to receive exercises, students and their attempts on the exercises with the resulting scores. From this data a user-item matrix has been established with the users being the students and the items being the exercises. Due to the little meta-data available the *Collaborative Filtering* algorithms of the RSs domain are used, because they are known to exploit the user-item matrix as opposed to *Content-based RSs* [1].

Intuitively the difficulty of a particular exercise for a particular student is based on how well similar students have performed on that particular exercise. Similarity measures are used to compute the similarity between two users or two items depending on whether a user-based or item-based approach is used. Several well known similarity measures have been tested such as the *Pearson Correlation*, *Tanimoto Coefficient* (aka *Jaccard*) and the *Loglikelihood* similarity measures. However due to the dichotomous nature of the data most of these similarity measures have proven to be ineffective into taking into account all the available information the data gives. Therefore an extended form of the *Tanimoto Coefficient* has been used that considers all possible cases of dichotomous data (e.g. correct / incorrect). The following formula computes the similarity for users u and v:

$$sim_{uv} = \frac{|\mathcal{C}_u \cap \mathcal{C}_v| + |\mathcal{I}_u \cap \mathcal{I}_v| - |\mathcal{C}_u \cap \mathcal{I}_v| - |\mathcal{I}_u \cap \mathcal{C}_v|}{|\mathcal{N}_u \cup \mathcal{N}_v|} \quad (1)$$

With $\mathcal{C}$ and $\mathcal{I}$ being the correct and incorrect answers respectively and $\mathcal{N}$ the total amount of attempts.

3 Observations

Experimental analysis shows that, when using the right combination of algorithms and similarity measures, the collaborative approach can provide an accurate classifier. As a way of verification, a second independent dataset has been used and experimental results show similar conclusions. Next to these findings some interesting observations have been found. The real-life dataset consisted of an uneven distribution where nearly 75% of the exercise attempts were correctly solved by students. There can be several reasons for this bias, e.g. the dataset could be of very low difficulty or the user-group could be of very high intelligence. Baseline algorithms, such as taking the most frequent action for a user or exercise, proved to be very effective with this data distribution. In the case of an uneven distribution, RSs were able to slightly improve upon the baselines, however obtained improvements were marginal in most cases. When testing more even distributions of correctly made exercise attempts (i.e. ratio of 50%), the baseline performance drops significantly, however the RSs we employed maintain a high prediction / recommendation accuracy, almost unaffected by the change in distribution.

Other observations show that the choice of similarity measure used can have a big influence on the accuracy of the collaborative filtering classifier. While most similarity measures are suited for data of either continuous or discrete form, only a few take into account the extra information provided by dichotomous data. The adapted Tanimoto Coefficient similarity measure has shown to consistently best other similarity measures. Moreover, algorithms based on this kind of similarity measure have shown to provide accurate classifications even when the neighbourhood of a user is fairly small (e.g. 5 users). From a practical point of view it has been shown in this dissertation that an item-based collaborative filtering approach is favoured over a user-based approach as there are ways to optimize neighbourhood-based similarity calculations when the items are much lower in amount than the users.

4 Conclusion

By experimental analysis the best-performing algorithm was an item-based approach with the modified *Tanimoto Coefficient* similarity measure. This algorithm has bested every baseline algorithm in every situation and shown to be an accurate classifier when predicting whether a particular student can pass a particular exercise. Having shown that the collaborative filtering can be used as an accurate classifier, we can now think of ways to use this information to improve the online learning experience of students using the works in this dissertation as part of a solution. If we go down that road, we will need to consider the pedagogical aspects of the use of a RS inside an educational environment. By being able to predict the difficulty for users on items, we can in future steps guide students towards new exercises tailored to their own needs creating an adaptive learning environment.

References

[1] Christian Desrosiers and George. Karypis. A comprehensive survey of neighborhood-based recommendation methods. In Ricci et al. [3], chapter 4, pages 107 – 144.

[2] Nikos Manouselis, Hendrik Drachsler, Riina Vuorikari, Hans Hummel, and Rob. Koper. Recommender systems in technology enhanced learning. In Ricci et al. [3], chapter 12, pages 387 – 415.

[3] Francesco Ricci, Lior Rokach, Bracha Shapira, and Paul B. Kantor, editors. *Recommender Systems Handbook*. Springer, 2011.

Studying Social Interactions using Swarm Robotics*

Irme M. Groothuis, Bijan Ranjbar-Sahraei

Department of Knowledge Engineering, Maastricht University

July 13, 2014

Abstract

Studying social behaviour has a very long history, and in behavioural sciences various models are suggested to study social behaviour. Typically agent-based simulators or real human experiments are used to study/validate these models. However, using robots, can help with the calibration of these models, where the physical environment is supposed to effect the interactions between robots. This research aims to implement an N-player Prisoner's Dilemma (PD). The N-player PD can be used to represent the behaviour of selfish and altruistic individuals in a society. Using robots and simulators one is able to model this more precisely and so one can build models of the evaluation of cooperation in society. This is needed as the physical environment is expected to influence behaviour. This paper mainly focuses on the scenario that robots wander around randomly in an unknown environment and according to the predefined strategies, either cooperation or defection, interact with each other whenever they meet each other by chance. This experiment is further extended to large groups of robots and both the microscopic and macroscopic behaviours of the group are studied. In this paper, a multi-robot framework is proposed in which robots can be used to study the social behaviour. This research seeks to answer to the following questions:

RQ 1 What is an appropriate two-robot experimental setting which can represent the simple PD problem?

RQ 2 How to extend the setting designed for the two-robot PD to a Swarm Robotic experiment?

RQ 3 What is the effect of structure of the environment on swarm outcomes?

RQ 4 Can robots be used to study social behaviours?

RQ 1 What is an appropriate two-robot experimental setting which can represent the simple PD problem? The chicken game is a appropriate setting to represent the prisoner's dilemma.

RQ 2 How to extend the setting designed for the two-robot PD to a Swarm Robotic experiment? The robots still play the game with only two players, but because of the random walk the robots can play with anyone.

To explore the possibilities for designing the proposed framework, different robotic platforms and available simulators are studied. The best set-up is chosen and an appropriate interaction strategy is implemented which captures the main properties of a chicken game, as a simple game-theoretical model, for a multi-robot system. Various experiments are conducted which are all implemented in the *Stage* simulator with robotic platform *Turtlebot*[1].

All experiments are are implemented by using the robot simulator *Stage* for the robotic platform Turtlebot. The robots walk around following a code that allows fro a simple random walk, while avoiding obstacles. Another program functions as a blackboard model: It collects the data from all the robots and stores them in one place, it then alerts a pair of robots whenever they are close enough to play the chicken game. The experiments tested the behaviour of the robots in different settings: four different environments were used and three different ratios between defectors and cooperators. Matlab was used to calculate the fitness of the robots. The fitness is the average speed of the robots.

*This thesis was prepared in partial fulfilment of the requirements for the Degree of Bachelor of Science in Knowledge Engineering, University of Maastricht, supervisors: Prof. Gerhard Weiss, Prof. Karl Tuyls, Phd. Candidate Bijan Ranjbar-Sahraei

[1]For a demonstration of the Turtlebots and how they were used in this project, see http://swarmlab.unimaas.nl/papers/bnaic-2014-demo/

RQ 3 What is the effect of environment structure on swarm outcomes?

In general we can conclude that the defectors seem to do well in the beginning, but after a certain period of time their scores decrease. The cooperators on the other hand, seem to need some time to get going, but afterwards their score steadily increases, allowing them to surpass the defectors. This can be explained in the following way: in the beginning, the defectors roam around freely, and the cooperators move around them, sometimes standing still to avoid them. After a while however, the defectors will get stuck in a collision with another defector, causing them to stop moving. When this has happened they become less of an obstacle for the cooperators, allowing them to roam around freely.

RQ 4 Can robots be used to study social behaviour? The experiment results confirm that the proposed framework can be helpful in validating and studying models of social behaviour.

Examining the effect of the physical environment has always been a challenge in studying social behaviours. Therefore, in this paper the application of swarm robotics to study social behaviour was introduced. The proposed framework uses ROS and Turtlebots as they are more convenient for this kind of research allowing to use the same code on both simulators and real robots. From the experiments conducted, it can be concluded that the environment does have an influence on the performance of the robots, as was suggested earlier.

Adaptive Learning Using the Exclusion Principle

Iris Monster [a] James M. McQueen [abc] Peter Desain [ac]

[a] *Radboud University Nijmegen*
[b] *Max Planck Institute for Psycholinguistics Nijmegen*
[c] *Donders Institute for Brain, Cognition and Behaviour Nijmegen*

1 Introduction

Studying foreign languages has become an important part of our educational system and society. Unfortunately, learning a foreign language takes a lot of time and effort. Therefore it is of great importance to find better and more efficient ways of learning or teaching a second language.

Researchers have found numerous ways to enhance vocabulary acquisition, including the exclusion principle. The exclusion principle can be used to learn new objects [4, 1], to train literacy skills [3], or to learn vocabulary. To learn new vocabulary one could be presented with a new word and the possible answers consist of the right answer (the key) and familiar distractors. Using the exclusion principle the learner can rule out these familiar options, because he or she knows they mean something else. This leaves the unfamiliar, right answer.

This study [2] tested whether the exclusion principle would enhance Dutch primary school children's ability to learn new English words in a flashcard application. The application implemented an adaptive, exclusion algorithm and a random, control algorithm. It was hypothesized that the children would learn faster (less repetitions) with the exclusion algorithm, gained more confidence whilst studying and performed better on a post-test.

2 Methods

2.1 Application & Testing Procedure

The application built for this experiment ran two sessions that both consisted of a learning phase, a short questionnaire and a post-test.

During the learning phase the child learned English words by being presented a Dutch word and four possible English translations. The child's task was to select the correct answer. If the child selected the right answer positive feedback was given and when the answer was wrong correcting feedback was presented. The program would repeat each word until completely mastered: correctly recognized at least 3 times. The exclusion condition presented distractors that were correctly recognized on previous trials and the control condition chose random distractors.

The questionnaire asked the child to rate his or her performance score before doing this task as a measure of confidence. The post-test presented each word one last time and the number of correct responses was measured. Here all distractors were random distractors.

The experiment was performed with 40 children in group 8 (age 11 to 13 years old). The children had to learn 30 words that they had not seen before and the words were all a singular noun of which the written English form does not resemble the Dutch translation. The wordlist was split in two equally difficult parts, one for each condition. The order of conditions and the combination with the wordlists were counterbalanced.

2.2 Analysis

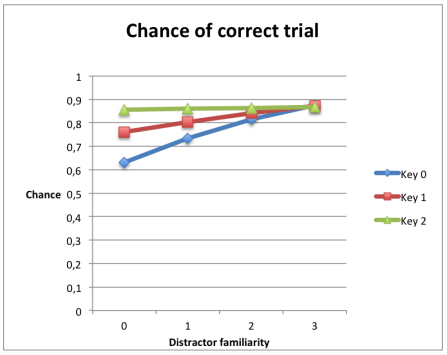
First the data was pre-processed to remove outliers, resulting in 35 remaining subjects. Then the two conditions were compared with a Wilcoxon Signed Ranks test (a non-parametric test to compare two related samples). Per condition the number of trials, the confidence score and the performance score on the post-test were compared.

After this analysis another test was done to compare the effect of familiar distractors. Each trial (of all subjects) was categorized by distractor familiarity (all with familiarity 3, familiarity 2 or higher, familiarity 1 or higher and familiarity 0 or higher). Using a logistic regression analysis a model was made to predict the outcome of a trial (right or wrong answer) based on the familiarity of the distractors and the key (0, 1 or 2).

3 Results

The first analysis to compare the two conditions showed no significant effect in the number of trials needed in the learning-phase, the confidence based on the questionnaire or the performance on the post-test. Which could be explained by the fact that after a number of trials the control condition started to resemble the exclusion condition, because the random distractors had also gained familiarity.

The second test did show a significant influence of the familiarity of the key and the distractors. The B-values given by the model were used to compute the chance of a correct trial based on the familiarity scores. The figure below shows the effect of distractor familiarity is most present when the key has not been recognized correctly before.



4 Conclusion

The conclusion of this study is that although there were no effects found in the main analysis and although the post-hoc test did not show an effect of learning itself, it is possible that the positive effect seen in the post-hoc analysis could mean that exclusitivity would also help with vocabulary learning.

References

[1] W. J. McIlvane and T. Stoddard. Acquisition of matching-to-sample performances in severe retardation: Learning by exclusion. *Journal of Mental Deficiency Research*, 25(1):33–48, March 1981.

[2] I. Monster, J. McQueen, and P. Desain. Adaptive learning using the exclusion principle. *Bachelor Thesis Artificial Intelligence, Radboud University Nijmegen*, 2014.

[3] M. M. Van Gogh, J. M. McQueen, and L. Verhoeven. Learning phonologically specific new words fosters rhyme awareness in dutch preliterate children. *In Press*, 2014.

[4] K. M. Wilkinson, C. Rosenquist, and W. J. McIlvane. Exclusion learning and emergent symbolic category formation in individuals with severe language impairments and intellectual disabilities. *The Psychological Record*, 59(2), 2009.

Rotation invariant feature extraction in the classification of galaxy morphologies

Steven Reitsma [a]

[a] *Radboud University Nijmegen*

Abstract

The Galaxy Zoo project is a crowdsourcing platform to classify the morphology of galaxies into different categories. Recently, the project set out a challenge to automatically predict these crowdsourced classifications using machine learning techniques. In this thesis, one of these machine learning techniques was explored and modified. This technique, designed by Coates et al. [2], works by learning features in an unsupervised manner using k-means. These features are rotation sensitive, but since there is no up or down in space, the system would ideally work rotation invariantly. Therefore, the system was modified to account for rotation sensitivity in an efficient manner by changing the distance metric that is used by the k-means algoritm. Results show that this improves the performance significantly by more than 5%: the regular method achieves a root mean squared error of 0.10789 while the rotation invariant method achieves a score of 0.10256.

1 Galaxy Zoo project

In July 2007, the Galaxy Zoo crowd sourcing project saw the light of day. Initially, the goal of this project was to classify the morphology of galaxies in three categories: elliptical galaxies, merged galaxies and spiral galaxies. Nowadays, there are more categories. The first instalment of the project used the SDSS[1] data set, which back then consisted of approximately a million galaxies. The classification of these morphologies is not automized. Instead, visitors of the Galaxy Zoo website can vote on the morphology class, given a photograph of a galaxy. During the first year of the project, fifty million classifications were received for almost a million galaxies.

1.1 Automation

While crowd sourcing the morphology classifications of galaxies is a great idea and much faster than having researchers classify the galaxies manually, the amount of data is just too vast to keep up with. Luckily, technological advancements have made it possible to automatically classify these galaxies, using machine learning techniques. This paper describes a modification of the system designed by Coates et al. [2], which is based on unsupervised feature extraction. The system is tested on a data set that the Galaxy Zoo project provided.

2 Data set

The data set consists of 61578 424x424 pixel training images and 79975 equal sized test images. All images contain a single galaxy and most of the galaxies are from a region called Stripe 82, which has been repeatedly imaged over the years [1]. For each of the training images, the distribution of votes from the Galaxy Zoo users is given. These consist of 37 real values between 0 and 1 [4].

[1] Sloan Digital Sky Survey

3 Unsupervised learning

Coates et al. [2] used much smaller images than the images in the Galaxy Zoo data set. To prevent incredibly long running times, the data set has to be shrunk. Firstly, images are cropped to 150x150 pixels, since the outer edge of the image contains mostly blackness. Secondly, images are reduced to a size of 15x15 pixels. As this does decrease performance, it would be interesting to see results for a decreased resize factor. Color data is kept at all times since the morphology of a galaxy and its temperature – and thus color – are related [3]. These 15x15 pixel images can now be fed into the pipeline that Coates et al. [2] designed. This pipeline consists of three phases. First, patches are randomly extracted from the images and clustered using k-means. The centroids can now be seen as representations of the features that are present in the images. Using the centroids, the activation for each image is calculated and pooled to result in a 12000-dimensional feature vector. These feature vectors are finally fed into a regressor.

4 Rotation invariance

In space there is no up or down. The rotation in which a galaxy was photographed has no effect on its morphology and thus we would ideally want the system to be rotation invariant. The most trivial way of handling this is to duplicate all training data and rotating it. This way, the classification system can also train on the rotated data which will result in a better performance were the system to encounter a test image that has features that have been seen before in the training set in a different rotation. However, this way comes with a major drawback: its complexity. Duplicating the data ρ times results in having to do ρ times more work. However, we also need to increase the amount of centroids in the k-means step, since we need to encode more features – the rotated ones. A better way is to modify the distance metric that k-means uses, since this removes the need for extra centroids. The modified distance metric works as follows. Firstly, every patch that is processed in the feature learning phase is rotated ρ times. Next, the Euclidean distance is computed between all rotations of every patch to all centroids. Then, for each patch, the rotation that resulted in the smallest distance to a centroid is selected and using this patch, the assigned centroid is updated. The results shown in the next section are based on this method.

5 Results and conclusion

For the regular pipeline, a root mean squared error of 0.10789 was achieved. Using the rotation invariant pipeline, the error was decreased by 5% to 0.10256. This difference is significant with a p-value $< .001$. Since the p-value indicates a significant difference between the root mean squared error of the regular pipeline and the error of the rotation invariant pipeline, and the difference is in the right direction, we can conclude that the rotation invariant pipeline is better at classifying the morphology of galaxies. Generalizing features (e.g. by making them rotation invariant) to detect variants of a patch seems to be good practice in certain domains, certainly for galaxy morphologies. Using the Kaggle challenge, the Galaxy Zoo team has shown that the performance of automatic classification is improving. Whether automatic classification is going to replace manual classification or crowd sourcing in its entirety any time soon remains to be seen.

References

[1] Kevork N Abazajian et al. The seventh data release of the Sloan Digital Sky Survey. *The Astrophysical Journal Supplement Series*, 182(2):543, 2009.

[2] A Coates, AY Ng, and H Lee. An analysis of single-layer networks in unsupervised feature learning. In *International Conference on Artificial Intelligence and Statistics*, pages 215–223, 2011.

[3] Iskra Strateva et al. Color separation of galaxy types in the Sloan Digital Sky Survey imaging data. *The Astronomical Journal*, 122(4):1861, 2001.

[4] Kyle W Willett et al. Galaxy Zoo 2: detailed morphological classifications for 304122 galaxies from the Sloan Digital Sky Survey. *Monthly Notices of the Royal Astronomical Society*, 2013.

Broad-Band Visually Evoked Potentials Towards Enhanced Speller BCIs

Jordy Thielen [ab] Philip van den Broek [ab] Jason Farquhar [ab]

Peter Desain [ab]

[a] *Radboud University Nijmegen, Comeniuslaan 4, 6525 HP Nijmegen*
[b] *Donders Center for Cognition, Montessorilaan 3, 6525 HR Nijmegen*

1 Introduction

A Brain-Computer Interface (BCIs) adds an additional channel to the brain that can be used for communication and control. This channel requires brain activity only, hence it is bypassing the peripheral nervous system. A BCI is commonly set up to allow sending multiple commands sequentially by recording the electroencephalogram (EEG), pre-processing, classification and feedback ([6]).

One application of BCI is the visual speller, in which a 6 by 6 grid is presented to the user, with one character in each cell ([3]). The user has to overtly attend the character of interest, while rows and columns are being flashed sequentially in random order. Whenever the a target is flashed, a so-called P300 response is evoked. A classifier is trained to recognize this response and hence can discriminate between targets and non-targets. The BCI can decode the user's intention by correlating the pattern of targets with the cell-specific stimulation patterns. This visual P300-based speller achieves an Information Transfer Rate (ITR) around 30 bits/min ($\approx$ 5 char/min). Although the advantage of using BCI over common input devices remains limited for healthy users, patients who have lost control over their muscles can benefit extremely.

To improve speed and robustness other paradigms have been explored. One of these is the Steady-State Visually Evoked Potential (SSVEP), which is a response evoked by periodic stimulation. Here, each character is encoded with a unique frequency. The attended frequency can be directly found back as increased power at the same frequency in brain activity. This method is called frequency-tagging and achieves an ITR around 70 bits/min ($\approx$ 12 char/min).

Another paradigm makes use of broad-band signals that evoke Broad-Band Visually Evoked Potentials (BBVEP). This paradigm has become known as noise-tagging as the underlying stimulation sequences have a noise-like characteristic. More specifically, the bit-sequences (i.e., on-off flashes) are called pseudo-random noise codes and exhibit low cross-correlations. BBVEP-based spellers have achieved an ITR up to 144 bits/min ($\approx$ 29 char/min) ([4]). We explored the BBVEP-based speller because of the improvement of noise-tagging over frequency-tagging and rows-columns.

2 Methods and Results

In a recent study we explored noise-tagging by using a set of Gold Codes as stimulation sequences, one for each cell ([5], [2]). We modulated the entire set by Phase Shift Keying, which limits the power in the lower part of the spectrum to reduce visual flicker. In addition, all bit-sequences then consist of a series of short (one one) and a long (two ones) flashes.

During training, a template matching classifier learns individual templates for each bit-sequence. An unseen testing trial can then be classified by maximizing the correlation between this trial and all templates. A downside of this approach is the need to learn a template for each class. This requires

averaging over 10 to 100 trials to increase the signal to noise ratio for each class, and results into inconvenient long training sessions.

To overcome this problem, we designed a method to predict templates without the need of extensive training: reconvolution ([1]). Each bit-sequence can be decomposed to the individual short and long flashes. If we know the transient response to these two events and assume a linear system, the EEG response to a bit-sequence can be obtained by summation of the transient responses, time-shifted according to the underlying bit-sequence. We can find an estimate of the transient responses by regression analysis. All codes contain only short and long flashes, hence it is possible to predict responses to unseen bit-sequences as well. By training on a few codes, reconvolution explains up to 82 percent of the variance for seen sequences, and up to 56 percent of the variance for unseen sequences.

A set of modulated Gold Codes consists of 67 separate codes. The speller only requires a selection of 36 codes, hence an optimization algorithm was defined that selects the subset of codes that contains the least maximum cross-correlation in their (predicted) responses. In addition, we optimized the allocation of this optimal subset of codes to the grid layout in such a way that responses to neighboring codes correlate least. Classification was performed by template matching as described above. With this pipeline a performance of 86 percent correct classification was reached on single-trials of 4.2 seconds, accounting for an ITR of 56 bits/min (n=12).

In the pipeline above, trials were recorded for 4.2 seconds always, on which the classifier made the decision. Some participants reached performances up to 100 percent, suggesting less data could have been sufficient to make a reliable prediction. By applying an early stopping algorithm, the classifier performed an output whenever it was confident. Moreover, the classifier was applied each 100 milliseconds, with a forced stop at 4.2 seconds. Confidence was based upon learned threshold margins defined as the difference between the best and second best correlation between single-trial and templates. On average classification performance remained at 86 percent, but now with trials of 3.2 seconds on average, yielding an ITR of 87 bits/min.

With the early stopping pipeline, the best participant achieved 97 percent correct classification with averaged trial lengths of 1.44 seconds. This corresponds to an ITR of 201 bits/min ($\approx$ 39 char/min).

3 Conclusion

By exploring noise-tagging as paradigm for stimulation, BBVEP-based BCI have proven to be very successful, outperforming the common P300- and SSVEP-based spellers. Because a generative model was defined, even responses to unseen bit-sequences can be predicted. This enabled us to fine-tune subject-specific parameters like optimal subset of codes, optimal layout allocation and stopping margins. By doing so, cross-correlations are minimized and discriminability between the templates is maximized.

References

[1] P. W. M. Desain and J. D. R. Farquhar. Method for processing a brain wave signal and brain computer interface, 2009. US Patent US20110251511.

[2] P. W. M. Desain, J. Thielen, P. L. C. van den Broek, and J. D. R. Farquhar. Brain computer interface using broadband evoked potentials, 2014. NL Patent 2013245.

[3] L. A. Farwell and E. Donchin. Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials. *Electroencephalography and clinical Neurophysiology*, 70(6):510–523, 1988.

[4] M. Spüler, W. Rosenstiel, and M. Bogdan. Online adaptation of a c-VEP brain-computer interface (BCI) based on error-related potentials and unsupervised learning. *PloS one*, 7(12):e51077, 2012.

[5] J. Thielen, P. L. C. van den Broek, J. D. R. Farquhar, and P. W. M. Desain. Broad Band Evoked Potentials for Fast and Robust Brain Computer Interfacing. submitted.

[6] Marcel van Gerven, Jason Farquhar, Rebecca Schaefer, Rutger Vlek, Jeroen Geuze, Anton Nijholt, Nick Ramsey, Pim Haselager, Louis Vuurpijl, Stan Gielen, and Peter Desain. The brain–computer interface cycle. *Journal of Neural Engineering*, 6(4):041001, 2009.

Finding Substitutions of Rare Earth Elements Using Publication Data

Kirill Tumanov

Department of Knowledge Engineering, Maastricht University,
P.O. Box 616, 6200 MD Maastricht

1 Introduction

Rare Earth Elements (REEs) attract a continuous interest of a wide range of studies and applications. Those elements possess specific chemical and physical properties beyond the characteristics of traditional elements [2, 1]. Nanotechnologies, being at the frontier of modern research are particularly eager to make use of REEs. However, since REEs are expensive to mine and process, developments in the field of their usage are limited.

Similar to the other industries, material scientists seek the productivity maximization of their compounds with the minimization of production costs. Therefore, a substantial effort is given to the search of more accessible substitutions of REEs. Obvious is a desire to preserve the unique properties of REEs.

It is clear that even with contemporary chemical and physical methods this search has a huge space of possible combinations. Thus, the studies on this topic rarely result in success. The aim of the present work is to demonstrate the automatic method facilitating the research in REEs' substitution finding.

In this study the possible substitutes are searched within a publication data in a field of nanotechnologies. The idea is to identify the substitution candidates from the past research in order to give an insight of what elements should be tested in the current studies. This information should significantly benefit the researchers in the field by limiting the substitutions' search space.

2 Method Description

Substitutes can be found using abstracts as a comprehensive but condensed source of information about a publication. It can be assumed that the substitutes might have a specific relationship with any of a REEs. Under this assumption abstracts are processed using modern Text Mining (TM) techniques [5].

Corpus of abstracts is preprocessed by a search of a verb phrase stem (VPS). When a stem is found, a sentence containing it is put away for the mining. A custom built TM system can be applied to the text fragments. The processing stages of the TM system include:

1. Sentence structure markup using StanfordNLP [3]
2. Co-reference resolution using ReVerb [4]
3. Relation extraction

The output relations should finally be filtered based on their possessing of at least one REE. The assumption above should ensure that a candidate substitute is a part of such a relation. Although, since search is done only for the specified VPS, other possible VPSs are ignored thus the corresponding substitutes are not identified.

The TM system may also be applied directly to the abstracts with the VPS of interest instead of the text fragments. This strategy results in a higher recall of substitutes, but in a lower precision. Note that as generally abstracts are kept short there is no critical increase of the processing time.

3 Results

In this work a corpus included more than 190.000 abstracts of articles in nanotechnology published form 1999 to 2003. Around 550 text fragments with the stem "substitut" were collected. In the end around 70 candidate relations were extracted and after manual revision of an "expert" the final results were obtained. Some of them are as shown in Table 1. Each "Substituent" is connected to a corresponding "Substance" by a generalized relation which may be read as "*used as a substitute in*". The elements and compounds of potential investment interest are marked in green.

Table 1: Exemplar substitutes of REEs and their chemical compounds

Substituent	Substance
Strontium	Lanthanum chromite Lanthanum manganites prepared by a sol-gel route Lanthanum Cobalt films prepared on Platinum
Titanium	Y_2Fe_{17} alloys
$Yttrium^{3+}$	$Zirconium^{4+}$
Molybdenum acid	Yttrium
Niobium	$Pr_8Fe_{86}B_6$ magnets
Neodymium	Barium
Samarium	$Nd_2Fe_{14}B$ Barium
Platinum	YPd_2B_2C

Similar techniques were applied to the abstracts directly as well. This resulted in an about 20% increase in a number of found substituents whilst the candidate substitutions number tripled and reached around 220 relations.

The results seem to be of high quality, but even so note that this information is only aimed to give a user an idea of possible research direction, and it by no means may be used to claim that those related elements are direct substitutes and clearly not in all cases. To be able to grasp the context around the relations it is still necessary to read the whole corresponding article, where the found relation was extracted from.

4 Conclusion

It was demonstrated that the application of advanced TM techniques may successfully provide researchers with cues in the process of REEs substitutes identification. Although the quality of the substitutes still requires an expert evaluation, the dramatically decreased search space largely saves the expert work in this evaluation. The described framework may also be easily extended or modified to support extraction of other relations of interest in the field of nanotechnology as well as in alternative domains.

References

[1] Nomenclature of inorganic chemistry. IUPAC Recommendations:51–52, 2005.

[2] Rare earths statistics and information. Technical report, United States Geological Survey, 6 2014.

[3] M C De Marneffe and C D Manning. Stanford typed dependencies manual. `http://nlp.stanford.edu/software/dependencies_manual.pdf`, 2008.

[4] A Fader, S Soderland, and O Etzioni. Identifying relations for open information extraction. *Proc. Conf. Empirical Methods in Nat. Lang. Proc.*, pages 1535–1545, 2011.

[5] R Feldman and J Sanger. *The text mining handbook*. Cambridge University Press, 2007.

Traffic Flow Optimization using Reinforcement Learning

Erwin Walraven

Algorithmics group, Delft University of Technology, The Netherlands

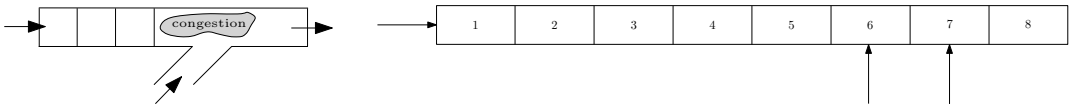
Abstract

Traffic congestion is an important problem that causes unnecessary delay, environmental pollution and more fuel consumption. In this thesis project we address this problem by proposing a new method to assign speed limits to highways. Our approach combines a macroscopic traffic flow model with Q-learning to generate speed limit policies, and we show that traffic predictions can be included in our method to regulate traffic flow more efficiently.

1 Problem Description

We study the problem of finding speed limits that can be assigned to a unidirectional highway to reduce congestion. An example highway is shown in Figure 1a, where the arrows indicate the direction of the vehicle flow. The rectangles represent a partitioning of the highway into sections. If the demand flow of the origin and the on-ramp is high, congestion may arise in the shaded region.

We aim to find a method that defines the speed limits that should be assigned to the sections, such that the global delay car drivers incur is minimized. A few additional constraints can be imposed. For example, speed limits should be increased and decreased smoothly and alternating sequences of speed limits should be prevented. Note that the schematic representation shown in Figure 1a can easily be generalized to highways with more sections and on-ramps, such as the highway shown in Figure 1b. The problem we defined occurs naturally in practice near on-ramps and road interchanges.



(a) Unidirectional highway with one on-ramp.

(b) Unidirectional highway consisting of eight sections.

Figure 1: Schematic representation of highways.

2 Learning Speed Limit Policies

We define a traffic flow optimization problem as a Markov Decision Process (MDP) [2]. Since we are unable to let our algorithm interact with a real highway, we use a macroscopic traffic flow model as a representation of the environment in which the agent assigns speed limits. The model we use is called METANET, and provides a set of formulas to compute speed, density and flow values for highway sections in discrete time steps [1]. It can be adapted such that speed limits are taken into account.

In the MDP formulation we consider the unidirectional highway shown in Figure 1b. The action space A consists of the speed limits 60, 80, 100 and 120 that can be assigned to section 3 to 6. We

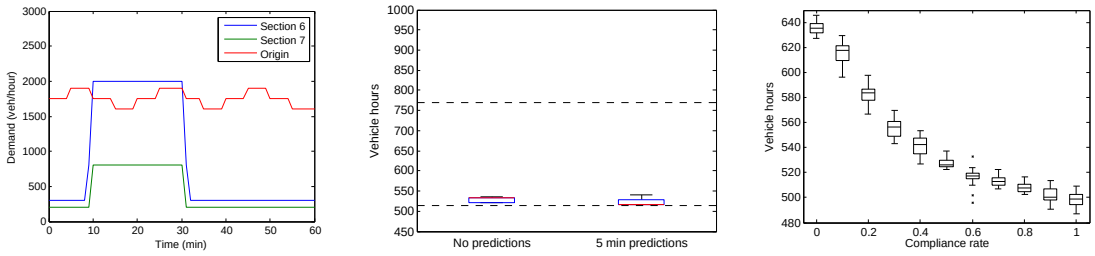
The full version of the thesis can be found in [3].

define a six dimensional state space S, where each state contains the last two speed limits that have been selected, as well as the speed values of section 4 to 7. The reward function represents a punishment received by the agent if speed limits are decreased too much, or if they are alternating. If there is no congestion, no punishment is given. In all other cases, the agent receives a punishment proportional to the number of vehicle hours that vehicles have made after the last speed limit assignment. The reward function ensures that speed limits are only activated if needed, and implicitly defines that the number of vehicle hours should be minimized.

We apply Q-learning to find a policy $\pi : S \rightarrow A$ for a given traffic scenario [4]. We enhanced this algorithm with artificial neural networks as a value function approximation technique, to handle a larger state space and it generalizes learning experiences. In addition to our work on speed limit policy learning, we made a first step to generalize this idea to multi-agent coordination of variable speed limits. In this approach, there is one agent associated with each highway section. We have also shown that knowledge regarding traffic flow regulation, represented by a policy, can be reused to learn policies for new traffic scenarios more efficiently. To investigate whether traffic predictions can be integrated in our method, we included model-based traffic predictions in the state description, and we have shown that policy quality improves.

3 Simulation Results

To evaluate our method, we defined several traffic scenarios, for which we have shown that the quality of the generated policies is close to lower bounds we were able to compute. Figure 2a shows the demand profile of one of the scenarios. The distribution of policy quality for this scenario is shown in Figure 2b. The horizontal lines represent the baseline and optimal number of vehicle hours, which is considered as a lower bound. It shows that the generated policies are close to the lower bound we found, and the policy quality is slightly better if model-based predictions are included in the state description. Additionally, we performed a case study involving a more realistic simulation of the A67 highway in



(a) Scenario demand profile. (b) Policy quality distribution. (c) A67 scenario simulation.

Figure 2: Simulation results.

The Netherlands. We derived a real traffic demand pattern for the A67 from NDW, the Dutch national database containing historical traffic data, and we performed a simulation in a microscopic simulation environment. We applied a speed limit policy during the simulations, for different compliance rates. The results are presented in Figure 2c, which shows that congestion can be reduced, and that small compliance rates already yield a significant improvement.

References

[1] A. Messner and M. Papageorgiou. METANET: A macroscopic simulation program for motorway networks. *Traffic Engineering & Control*, 31(8-9):466–470, 1990.

[2] M.L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, Inc., New York, NY, USA, 1st edition, 1994.

[3] E. Walraven. Traffic Flow Optimization using Reinforcement Learning. Master's thesis, Delft University of Technology, 2014.

[4] C.J.C.H. Watkins and P. Dayan. Q-learning. *Machine Learning*, 8(3-4):279–292, 1992.

Topics

AI for Ambient Intelligence, 151, 161
AI for E-commerce, 33
AI for Games & Entertainment, 9, 119, 137, 185, 187, 193
AI for Robotics, 1, 73, 103, 181, 193, 205
AI for Software, 33
AI in Art, 57
AI in Education, 137, 193, 203, 207
AI in Medicine, 147, 177, 193
AI in Music, 57

Classification, 1, 65, 133, 141, 161, 193, 199, 211
Clustering, 65, 161, 199, 211
Cognitive Modeling, 73, 137, 149, 151, 199, 211

Embodied Artificial Intelligence, 1, 73, 149, 181
Evolutionary Algorithms, 129, 157, 167, 173

Hybrid Intelligent Systems, 149

Intelligent Agents & Multi-Agent Systems, 33, 73, 95, 103, 137, 149, 151, 153, 155, 159, 161, 165, 179, 181, 191, 205

Knowledge Discovery and Data Mining, 141, 201, 213
Knowledge Representation, 17, 25, 33, 111, 131, 141, 143, 145, 147, 189, 191
Knowledge-based Systems, 145, 189, 191

Logic in AI, 17, 25, 81, 88, 131, 141, 143
Logic Programming, 189

Machine Learning, 1, 41, 49, 133, 161, 177, 193, 199, 201, 211

Natural Artificial Intelligence, 73, 149, 181
Natural Language Processing, 65, 189, 213
Neural Networks, 49, 193

Ontologies, 111, 141, 175

Pattern Discovery, 161, 193, 201
Planning & Scheduling, 129, 153, 163, 169, 179

Reasoning under Uncertainty, 169, 189
Recommender Systems, 161, 203
Reinforcement Learning, 139, 187, 215

Search & Retrieval, 9, 65, 119, 129, 171, 189
Semantic Web-Techniques & Technologies, 111, 135, 145, 147, 175, 189
Swarm Intelligence, 103, 205

Verification & Validation, 103, 151, 205
Visualization, 133, 153, 179, 189

Author Index

Adriaensen, Steven, 129

Backer, Niels, 1
Bakkes, Sander, 185
Baniasadi, Zohreh, 88
Baroni, Pietro, 131
Barton, Michael, 9
Bauer, Pavol, 167
Bazoobandi, Hamid, 135
Beauxis-Aussalet, Emma, 133
Beek, Wouter, 135
Beinema, Tessa, 199
Berkel, Bas van, 201
Boella, Guido, 131
Bonhof, Andre, 189
Booth, Richard, 17
Bosch, Antal van den, 65
Bosman, Peter, 153, 157, 167, 173
Bosse, Tibor, 137
Brinke, Steven te, 119
Broek, Philip van den, 211
Broekens, Joost, 149, 181
Brys, Tim, 129, 139

Caminada, Martin, 25
Cerutti, Federico, 131
Claassen, Tom, 177
Claessen, Felix, 153

De Causmaecker, Patrick, 141
De Hauwere, Yann-Michael, 193
De Wannemacker, Stefan, 141
Dekker, Andre, 147
Desain, Peter, 207, 211
Devillé, Maarten, 203
Devriendt, Floris, 203
Dietrich, Dietmar, 73
Doder, Dragan, 143
Dong, Huimin, 81
Dotch, Ron, 199
Driessens, Kurt, 49

Endriss, Ulle, 155

Farquhar, Jason, 211

Ghanavati, Sepideh, 33
Giacomin, Massimiliano, 131
Gomez Marulanda, Felipe, 193
Groot, Perry, 41, 177
Groothuis, Irme, 103, 205
Grootjen, Franc, 199
Guéret, Christophe, 145

Hardman, Lynda, 133
Hashemi, Vahid, 155
Herik, Jaap van den, 171
Heskes, Tom, 41, 177
Hindriks, Koen, 181
Hoang Luong, Ngoc, 157
Hoeksma, Jesper, 189
Hoekstra, Rinke, 145
Hoogland, Jasper, 159
Huang, Zhisheng, 147
Huijbrechts, Bas, 189
Huisman, Bob, 179

Jonker, Catholijn, 149

Kamphorst, Bart, 151
Kauschinger, Manuel, 49
Klein, Michel, 151
Klos, Tomas, 179
Kooijker, Bas, 57
Kudenko, Daniel, 139
Kunneman, Florian, 65

La Poutré, Han, 153, 157, 159
Laarhuis, Jan, 189
Lagatie, Ruben, 203
Liefers, Bart, 153, 159

Maggiore, Giuseppe, 161
Mavromoustakos Blom, Paris, 185
McQueen, James, 207
Meertens, Roland, 187
Michels, Steffen, 189
Monster, Iris, 207

Neerincx, Mark, 181
Nijs, Frits de, 163
Novak, Peter, 189

Nowé, Ann, 129, 139, 193, 203

Oliehoek, Frans, 169
Oren, Nir, 25

Pauwels, Eric, 153
Peters, Markus, 41
Plaat, Aske, 149, 161, 171
Podlaszewski, Mikołaj, 17, 191
Polevoy, Gleb, 165
Provoost, Simon, 137

Ranjbar-Sahraei, Bijan, 103, 205
Reitsma, Steven, 209
Rietveld, Laurens, 135, 145
Rodrigues, Sílvio, 167
Roijers, Diederik, 169, 185
Roos, Nico, 95, 175
Ruijl, Ben, 171

Sadakata, Makiko, 57
Sadowski, Krzysztof, 173
Sandberg, Jacobijn, 111
Santos, Carlos, 161
Schaat, Samer, 73
Schadd, Frederik, 175
Scharpff, Joris, 169
Scheepens, Roeland, 189
Schlobach, Stefan, 135, 145
Schreiber, Guus, 111
Sokolova, Elena, 177
Spaan, Matthijs, 163, 169
Sun, Xin, 81, 88
Sun, Zhenglong, 95

Taylor, Matthew, 139
Teitsma, Marten, 111
Ten Teije, Annette, 147
Thielen, Jordy, 211
Thierens, Dirk, 173
Torre, Leon van der, 131
Trajanovski, Stojan, 165
Tumanov, Kirill, 213
Tuyls, Karl, 103

Uiterwijk, Jos, 9

Van Harmelen, Frank, 147
Velikova, Marina, 189
Vermaseren, Jos, 171
Villata, Serena, 131
Visser, Arnoud, 1
Vrancx, Peter, 193, 203

Walraven, Erwin, 215
Weerdt, Mathijs de, 163, 165, 169
Weiss, Gerhard, 103
Wevers, Lesley, 119
Whiteson, Shimon, 169
Wielemaker, Jan, 135
Wielinga, Bob, 111
Wilson, Michel, 179
Wissen, Arlette van, 151
Witteveen, Cees, 179
Wolf, Ketter, 41
Woltran, Stefan, 143
Wu, Yining, 191

Xu, Junchao, 181

Zee, Marc van, 33
Zhou, Shuwen, 88