

# Local Fourier analysis of multigrid for the curl-curl equation

*Tim Boonen*

*Jan Van lent*

*Stefan Vandewalle*

*Report TW 484, December 2006*



Katholieke Universiteit Leuven  
Department of Computer Science  
Celestijnenlaan 200A – B-3001 Heverlee (Belgium)

# Local Fourier analysis of multigrid for the curl-curl equation

*Tim Boonen\**  
*Jan Van lent*  
*Stefan Vandewalle*

*Report TW 484, December 2006*

Department of Computer Science, K.U.Leuven

## Abstract

We present a local Fourier analysis of multigrid methods for the two-dimensional curl-curl formulation of Maxwell's equations. Both the hybrid smoother proposed by Hiptmair and the overlapping block smoother proposed by Arnold, Falk and Winther are considered. The key to our approach is the identification of two-dimensional eigenspaces of the discrete curl-curl problem by decoupling the Fourier modes for edges with different orientations. Our analysis allows to quantify the smoothing properties of the considered smoothers and the convergence behavior of the considered multigrid methods. Additionally, we identify the Helmholtz splitting in Fourier space. This allows to recover several well known properties in Fourier space, such as the commutation properties of the classical Nédélec prolongator and the equivalence of the curl-curl operator and the vector Laplacian for divergence-free vectors. We show how the approach used in this paper can be generalized to two- and three-dimensional problems in  $H(\text{curl})$  and  $H(\text{div})$  and to other types of regular meshes.

**Keywords :** multigrid, curl-curl equation, local Fourier analysis

**AMS(MOS) Classification :** 65N55, 65N12, 65F10

---

\*Tim Boonen is research assistant of the Fund for Scientific Research - Flanders (Belgium) (F.W.O.-Vlaanderen).

# LOCAL FOURIER ANALYSIS OF MULTIGRID FOR THE CURL-CURL EQUATION

TIM BOONEN\*, JAN VAN LENT , AND STEFAN VANDEWALLE

**1. Introduction.** The convergence of iterative methods for linear systems that arise from the discretization of a partial differential equation (PDE) can often be studied in a model problem setting by means of Fourier modes. In a local Fourier analysis (LFA) or local mode analysis, an infinite, regular grid is considered and boundary conditions are not taken into account. This type of analysis was introduced in [5]. It has been applied successfully to iterative solvers for several types of PDEs, such as the diffusion equation, the convection-diffusion equation, the Helmholtz equation, the biharmonic equation, the Stokes equations, the Oseen equations and elasticity. For comprehensive surveys, we refer the reader to [13, 15, 14].

In this paper, we develop a LFA for some multigrid methods for the curl-curl equation

$$\operatorname{curl}(\alpha \operatorname{curl}(\vec{A})) + \beta \vec{A} = \vec{J}, \quad \alpha \neq 0. \quad (1.1)$$

This equation arises in magnetostatic problems and in time-domain and frequency-domain approaches for solving time-dependent electromagnetic problems, such as the eddy current problem and electromagnetic wave scattering. We restrict ourselves to the finite element discretization of the two-dimensional curl-curl equation with constant complex material parameters  $\alpha$  and  $\beta$ . Appropriate scaling allows to consider only the case  $\alpha = 1$ . For the discretization, standard first order edge elements on a regular quadrilateral grid and conforming in  $H(\operatorname{curl})$  will be used. This leads to a linear system of the following form, with the degrees of freedom related to the edges of the grid:

$$\left( \frac{1}{h^2} K_{cc} + \beta M \right) x = b. \quad (1.2)$$

The stiffness matrix  $K_{cc}$  has a large kernel, which is essentially equal to the range of the discrete gradient operator  $G$ . The splitting of the vector space  $\mathcal{E}$  of the degrees of freedom, induced by  $G$ , is the discrete analogue of the Helmholtz splitting of  $H(\operatorname{curl})$  (see [9, 4]). This splitting reads, with  $C$  the discrete curl operator:

$$\mathcal{E} = \operatorname{ran}(G) \oplus \operatorname{ran}(C^T), \quad CG \equiv 0. \quad (1.3)$$

Typically, the smoothing properties of classical multigrid smoothers, as described in for instance [13, 12], are very poor in  $\operatorname{ran}(G)$  when applied to (1.2). More involved smoothers are needed here, such as the so-called hybrid smoother proposed in [7] or the overlapping block smoother proposed in [1]. Also, in order for multigrid to be effective, the prolongators  $P_e$  for the degrees of freedom related to the edges should satisfy a commutation constraint of the following form:

$$P_e G_c = G_f P_n. \quad (1.4)$$

Here,  $P_n$  denotes a prolongator for degrees of freedom related to the nodes and  $G_f$  and  $G_c$  denote the gradient operators on the corresponding fine and coarse level of the

---

\*Corresponding author. Department of Computer Science, Katholieke Universiteit Leuven, Celestijnenlaan 200A, B-3001 Leuven, Belgium ([tim.boonen@cs.kuleuven.be](mailto:tim.boonen@cs.kuleuven.be)). Research Assistant of the Fund for Scientific Research - Flanders (Belgium) (F.W.O.-Vlaanderen)

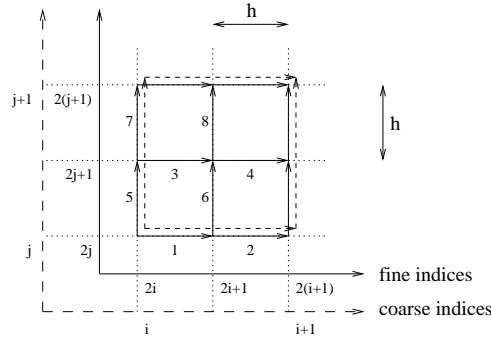


FIG. 2.1. Fine and coarse rectangular grids. The coarse nodes are the fine nodes with even indices. The dotted arrows represent the coarse edges. The numbers indicate the colors of the fine edges in a 8-color scheme.

hierarchy. This essentially guarantees that the structure of  $\mathcal{E}$ , represented by (1.3), is transferred in an appropriate way to the coarse levels of the grid hierarchy. We refer to [7] for further details.

Despite the development of several algebraic multigrid algorithms based on the algorithms of [7] and [1] (for instance [3, 2, 8, 10]), a Fourier analysis of these algorithms is not available yet. In fact, the nature of the degrees of freedom for discretizations of (1.1), which are related to edges instead of nodes, makes a straightforward application of classical Fourier analysis techniques impossible. The only existing related result is, to the knowledge of the authors, the derivation of the eigenvalues and eigenvectors of some curl-related operators in [6].

This paper is organized as follows. Section 2 introduces the definitions and notations used throughout this paper. In section 3, the classical Fourier modes are generalized for edge-based discretizations. In section 4, the Helmholtz splitting is identified in the Fourier domain. In section 5, the LFA of the curl-curl and mass operators is developed. In section 6, several classical multigrid smoothers, the hybrid smoother of [7] and the overlapping block smoother of [1] are analyzed and their smoothing properties are quantified. Aliasing phenomena are analyzed in section 7.1. In sections 7.2 and 7.3, the restriction and prolongation are studied. Finally, the results of the previous sections are combined in section 8 into a LFA of the 2-level multigrid cycle. Section 9 explains the numerical tests used to verify the analytical results presented in this paper. In section 10, it is indicated how the approach used in this paper can be generalized to other problems. We end with a conclusion in section 11.

**2. Definitions and notations.** We consider an infinite, two-dimensional regular quadrilateral grid  $G_h$  with grid size  $h$  (see Figure 2.1):

$$G_h := \{\mathbf{x} = h\mathbf{k} \mid \mathbf{k} \in \mathbb{Z}^2\}. \quad (2.1)$$

The degrees of freedom of a typical discretization of the curl-curl equation are associated with the oriented edges of the grid. The orientation of the edges is chosen as indicated in Figure 2.1. The horizontal and vertical orientation will be referred to by the subscripts  $\cdot_1$  and  $\cdot_2$  respectively.

With respect to the grid  $G_h$ , a node  $\mathbf{x} = (k_1, k_2)h$  will be referred to as  $(k_1, k_2)$ . Edges and faces will be referred to using their center points. For instance,  $(\frac{r_1+s_1}{2}, \frac{r_2+s_2}{2})$  refers to the edge connecting the nodes  $(r_1, r_2)$  and  $(s_1, s_2)$ . Further on, we will also

need a *displaced index set*  $I(\Delta_1, \Delta_2)$ . With  $\Delta_1, \Delta_2 \in \mathbb{R}$ , this set is defined as

$$I(\Delta_1, \Delta_2) := \{(k_1 + \Delta_1, k_2 + \Delta_2) \mid k_1, k_2 \in \mathbb{Z}\}.$$

This allows to define the sets of the indices of all nodes, horizontal and vertical edges and faces of  $G_h$  as  $N := I(0, 0)$ ,  $E_1 := I(\frac{1}{2}, 0)$ ,  $E_2 := I(0, \frac{1}{2})$  and  $F := I(\frac{1}{2}, \frac{1}{2})$  respectively. The set of the indices of *all* edges is defined as  $E := E_1 \cup E_2$ .

The *vector spaces* of the degrees of freedom related to the nodes, edges and faces of  $G_h$  will be denoted  $\mathcal{N}$ ,  $\mathcal{E}$  and  $\mathcal{F}$ . Note that these vector spaces have infinite dimension, since the number of nodes, edges and faces in  $G_h$  is infinite.

We define the following *evaluation operator* in a point  $\mathbf{k} = (k_1, k_2)$ :

$$\cdot_{\mathbf{k}} : f(\mathbf{x}) \mapsto f(\mathbf{x})_{\mathbf{k}} := f(hk_1, hk_2). \quad (2.2)$$

The vector of the evaluations of a function  $f$  in the nodes represented by the index set  $S$  will be denoted  $f(x)_S$ . This allows to represent vectors belonging to  $\mathcal{N}$  and  $\mathcal{E}$  by means of functions as  $f(x)_N$  and  $f(x)_E$ . Such vectors will often be referred to as grid functions, as they consist of a value at each point of a certain grid, specified by the subscript. The operator  $\cdot_{\mathbf{k}}$  will also be used as a selection operator, when applied to a vector. For example,  $(f(x)_S)_{\mathbf{k}} = f(\mathbf{x})_{\mathbf{k}}$ , if  $\mathbf{k} \in S$ .

We consider discrete edge-based operators  $L^e$  on  $G_h$  characterized by *difference stencils* with constant coefficients. An important difference with stencils for node-based operators is that the orientation of the edges has to be taken into account. This implies that *two* difference stencils are needed to represent a 2D edge-based operator on a quadrilateral grid, one for the horizontal and one for the vertical edges. With  $T = I(0, 0) \cup I(\frac{1}{2}, \frac{1}{2})$ , the application of a discrete edge operator  $L^e$  to a vector  $w(x)_E$  is defined as:

$$(L^e w(x)_E)_{\mathbf{k}} := \begin{cases} \sum_{\mathbf{t} \in T} s_{\mathbf{t}}^{[1]} w((\mathbf{k} + \mathbf{t})h) & \text{if } \mathbf{k} \in E_1, \\ \sum_{\mathbf{t} \in T} s_{\mathbf{t}}^{[2]} w((\mathbf{k} + \mathbf{t})h) & \text{if } \mathbf{k} \in E_2. \end{cases} \quad (2.3)$$

The coefficients of the horizontal and vertical stencil are denoted  $s_{\mathbf{t}}^{[1]}$  and  $s_{\mathbf{t}}^{[2]}$  respectively. The stencils corresponding to (2.3) will be represented as follows, for instance for the vertical stencil:

$$\begin{bmatrix} \vdots & & \vdots & & \vdots & & \\ \cdots & \circ & s_{-\frac{1}{2}, \frac{1}{2}}^{[2]} & \bullet & s_{\frac{1}{2}, \frac{1}{2}}^{[2]} & \circ & \cdots \\ & s_{-1, 0}^{[2]} & & s_{0, 0}^{[2]} & & s_{1, 0}^{[2]} & \\ \cdots & \circ & s_{-\frac{1}{2}, -\frac{1}{2}}^{[2]} & \bullet & s_{\frac{1}{2}, -\frac{1}{2}}^{[2]} & \circ & \cdots \\ & \vdots & & \vdots & & \vdots & \end{bmatrix} \quad (2.4)$$

The nodes of the edge to which the result of the stencil operator is assigned, are indicated by filled circles, and all other nodes by open circles. The two stencils corresponding to (1.2) read

$$\frac{1}{h^2} \begin{bmatrix} \circ & -1 & \circ \\ -1 & & 1 \\ \bullet & 2 & \bullet \\ 1 & & -1 \\ \circ & -1 & \circ \end{bmatrix} + \beta \begin{bmatrix} \circ & \frac{1}{6} & \circ \\ 0 & & 0 \\ \bullet & \frac{2}{3} & \bullet \\ 0 & & 0 \\ \circ & \frac{1}{6} & \circ \end{bmatrix} \quad (2.5)$$

for the horizontal stencils, and

$$\frac{1}{h^2} \begin{bmatrix} \circ & -1 & \bullet & 1 & \circ \\ -1 & & 2 & & -1 \\ \circ & 1 & \bullet & -1 & \circ \end{bmatrix} + \beta \begin{bmatrix} \circ & 0 & \bullet & 0 & \circ \\ \frac{1}{6} & & \frac{2}{3} & & \frac{1}{6} \\ \circ & 0 & \bullet & 0 & \circ \end{bmatrix}. \quad (2.6)$$

for the vertical stencils. To simplify the notation, the analysis will be done for  $(K_{cc} + h^2\beta M)x = h^2b$  with the factor  $h^2$  integrated into the material parameter as

$$\beta_h := h^2\beta.$$

**3. Fourier modes.** Contrary to the case of node-based operators, the vectors of the evaluations of

$$\varphi(\theta, \mathbf{x}) := e^{i\theta \cdot \mathbf{x}/h} := e^{i\frac{\theta_1 x_1 + \theta_2 x_2}{h}} \quad \text{with } \theta, \mathbf{x} \in \mathbb{R}^2 \quad (3.1)$$

in the centers of the edges are *not* the eigenvectors of discrete edge operators  $L^e$  of the form (2.3). In fact, the application of  $L^e$  to  $\varphi(\theta, \mathbf{x})_E$  amounts to a *different* scaling for the horizontal and vertical components:

$$\begin{aligned} (L^e \varphi(\theta, \mathbf{x})_E)_\mathbf{k} &= \begin{cases} \sum_{\mathbf{t} \in T} s_{\mathbf{t}}^{[1]} e^{i\theta \cdot (\mathbf{k} + \mathbf{t})} & \text{if } \mathbf{k} \in E_1 \\ \sum_{\mathbf{t} \in T} s_{\mathbf{t}}^{[2]} e^{i\theta \cdot (\mathbf{k} + \mathbf{t})} & \text{if } \mathbf{k} \in E_2 \end{cases} \\ &= \varphi(\theta, \mathbf{x})_\mathbf{k} \cdot \begin{cases} \sum_{\mathbf{t} \in T} s_{\mathbf{t}}^{[1]} e^{i\theta \cdot \mathbf{t}} & \text{if } \mathbf{k} \in E_1 \\ \sum_{\mathbf{t} \in T} s_{\mathbf{t}}^{[2]} e^{i\theta \cdot \mathbf{t}} & \text{if } \mathbf{k} \in E_2. \end{cases} \end{aligned} \quad (3.2)$$

This decoupling is a direct consequence of the fact that the stencils for the horizontal and vertical edges are different. Consequently,  $\varphi(\theta, \mathbf{x})_E$  cannot be used in the LFA. Instead, we define  $F_e(\theta)$  to be the two-dimensional space of all edge-based grid functions  $\phi(\theta)_E$  of the form

$$\phi(\theta)_E = \alpha_1 \phi_1(\theta)_E + \alpha_2 \phi_2(\theta)_E, \quad \alpha_1, \alpha_2 \in \mathbb{C}, \quad (3.3)$$

with

$$\phi_1(\theta)_E := \begin{bmatrix} \varphi(\theta, \mathbf{x})_{E_1} \\ 0_{E_2} \end{bmatrix}, \quad \phi_2(\theta)_E := \begin{bmatrix} 0_{E_1} \\ \varphi(\theta, \mathbf{x})_{E_2} \end{bmatrix}. \quad (3.4)$$

Note that this amounts to a decoupling of the horizontal and vertical components, equivalent to:

$$\phi(\theta)_\mathbf{k} = \begin{cases} \alpha_1 \varphi(\theta, \mathbf{x})_\mathbf{k} & \text{if } \mathbf{k} \in E_1 \\ \alpha_2 \varphi(\theta, \mathbf{x})_\mathbf{k} & \text{if } \mathbf{k} \in E_2, \end{cases} \quad \text{with } \alpha_1, \alpha_2 \in \mathbb{C}. \quad (3.5)$$

Further, we shall write (3.3) more compactly as

$$\phi(\theta)_E = \Phi(\theta) \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix}, \quad \text{with } \Phi(\theta) := [ \phi_1(\theta)_E \quad \phi_2(\theta)_E ]. \quad (3.6)$$

$\Phi(\theta)$  will play the same role in the LFA of edge-based problems as the vector  $\varphi(\theta, \mathbf{x})_N$  in the LFA of node-based problems. This role is based on the following properties.

**THEOREM 3.1** (Invariant subspace). *For all  $\theta \in \mathbb{R}^2$ ,  $F_e(\theta)$  is invariant for a discrete edge operator  $L^e$  of the form (2.3).*

*Proof.* The application of  $L^e$  to  $\phi(\theta)_E = \Phi(\theta)[\alpha_1 \ \alpha_2]^T \in F_e(\theta)$  yields for a horizontal edge  $\mathbf{k} \in E_1$ :

$$\begin{aligned} (L^e \phi(\theta)_E)_{\mathbf{k}} &= \sum_{\mathbf{t} \in T} s_{\mathbf{t}}^{[1]} \phi(\theta)_{\mathbf{k}+\mathbf{t}} \\ &= \alpha_1 \sum_{\mathbf{t} \in I(0,0)} s_{\mathbf{t}}^{[1]} e^{i\theta \cdot (\mathbf{k}+\mathbf{t})} + \alpha_2 \sum_{\mathbf{t} \in I(\frac{1}{2}, \frac{1}{2})} s_{\mathbf{t}}^{[1]} e^{i\theta \cdot (\mathbf{k}+\mathbf{t})} \\ &= e^{i\theta \mathbf{k}} \left( \alpha_1 \sum_{\mathbf{t} \in I(0,0)} s_{\mathbf{t}}^{[1]} e^{i\theta \cdot \mathbf{t}} + \alpha_2 \sum_{\mathbf{t} \in I(\frac{1}{2}, \frac{1}{2})} s_{\mathbf{t}}^{[1]} e^{i\theta \cdot \mathbf{t}} \right). \end{aligned}$$

A similar result holds for the vertical edges. So, the combined result

$$L^e \phi(\theta)_E = L^e \Phi(\theta) \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix} = \Phi(\theta) \begin{bmatrix} \beta_1 \\ \beta_2 \end{bmatrix}, \quad (3.7)$$

with

$$\begin{bmatrix} \beta_1 \\ \beta_2 \end{bmatrix} = \begin{bmatrix} \sum_{\mathbf{t} \in I(0,0)} s_{\mathbf{t}}^{[1]} e^{i\theta \cdot \mathbf{t}} & \sum_{\mathbf{t} \in I(\frac{1}{2}, \frac{1}{2})} s_{\mathbf{t}}^{[1]} e^{i\theta \cdot \mathbf{t}} \\ \sum_{\mathbf{t} \in I(\frac{1}{2}, \frac{1}{2})} s_{\mathbf{t}}^{[2]} e^{i\theta \cdot \mathbf{t}} & \sum_{\mathbf{t} \in I(0,0)} s_{\mathbf{t}}^{[2]} e^{i\theta \cdot \mathbf{t}} \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix} \quad (3.8)$$

belongs to  $F_e(\theta)$  as well.  $\square$

The  $2 \times 2$  matrix in (3.8) is the representation of the edge operator  $L^e$  in  $F_e(\theta)$ . It will be denoted  $L^e(\theta)$ .

**THEOREM 3.2** (Frequency aliasing). *All spaces  $F_e(\theta_1 + 2\pi a, \theta_2 + 2\pi b)$  with  $a, b \in \mathbb{Z}$ , are identical, for any  $\theta_1, \theta_2 \in \mathbb{R}$ .*

*Proof.* It is easy to check that the basis (3.6) of  $F_e(\theta_1 + 2\pi a, \theta_2 + 2\pi b)$  and  $F_e(\theta_1, \theta_2)$  are related as

$$\Phi(\theta_1 + 2\pi a, \theta_2 + 2\pi b) = \Phi(\theta_1, \theta_2) \begin{bmatrix} (-1)^a & 0 \\ 0 & (-1)^b \end{bmatrix}.$$

$\square$

Theorem 3.2 shows that it is sufficient to consider any frequency interval of length  $2\pi$  for  $\theta_1$  and  $\theta_2$ . In this paper, the two-dimensional frequency interval  $\Theta := [-\pi/2, 3\pi/2)^2$  will be studied. Note that  $\Theta$  is different from the frequency interval  $[-\pi, \pi)^2$  which is usually considered in a LFA (see for instance [13]). This choice will lead to a considerable simplification of the LFA for edge-based problems. The low and high frequency interval will be denoted

$$\Theta^{\text{low}} := [-\pi/2, \pi/2)^2 \quad \Theta^{\text{high}} := \Theta \setminus \Theta^{\text{low}}.$$

We define the harmonics of a low frequency  $\theta = (\theta_1, \theta_2) \in \Theta^{\text{low}}$  as:

$$\theta^{(a,b)} = (\theta_1 + a\pi, \theta_2 + b\pi), \quad a, b \in \{0, 1\}. \quad (3.9)$$

For the LFA of the smoother of [1], the prolongation, the restriction and the multigrid cycle, an 8-color variant of (3.5)

$$\psi(\theta)_{\mathbf{k}} = \alpha_i \varphi(\theta, \mathbf{x})_{\mathbf{k}}, \quad \alpha_i \in \mathbb{R} \quad \text{if } \mathbf{k} \in E_i^8 \quad (3.10)$$

will be needed. Here,  $E_i^8$ , with  $1 \leq i \leq 8$ , denotes the set of the indices of all edges with color  $i$ . The coloring scheme is indicated in Figure 2.1. The following Lemma

will be used to write a grid function of the form (3.10) as a linear combination of vectors of the form (3.6). This Lemma extends Remark 4.4.2 in [13].

LEMMA 3.3. *Let  $\Delta = (\Delta_1, \Delta_2) \in \mathbb{R}^2$  be a fixed displacement and consider a grid function satisfying*

$$\psi(\theta)_{k_1+\Delta_1, k_2+\Delta_2} = \begin{cases} \alpha_{00}\varphi(\theta, \mathbf{x})_{k_1+\Delta_1, k_2+\Delta_2}, & \alpha_{00} \in \mathbb{C} & \text{if } k_1 \text{ and } k_2 \text{ even} \\ \alpha_{10}\varphi(\theta, \mathbf{x})_{k_1+\Delta_1, k_2+\Delta_2}, & \alpha_{10} \in \mathbb{C} & \text{if } k_1 \text{ odd and } k_2 \text{ even} \\ \alpha_{01}\varphi(\theta, \mathbf{x})_{k_1+\Delta_1, k_2+\Delta_2}, & \alpha_{01} \in \mathbb{C} & \text{if } k_1 \text{ even and } k_2 \text{ odd} \\ \alpha_{11}\varphi(\theta, \mathbf{x})_{k_1+\Delta_1, k_2+\Delta_2}, & \alpha_{11} \in \mathbb{C} & \text{if } k_1 \text{ and } k_2 \text{ odd.} \end{cases}$$

Then, this grid function can be written as a linear combination of the grid functions  $\varphi(\theta, \mathbf{x})_{N+\Delta}$  for the harmonic frequencies of  $\theta$  given in (3.9):

$$\psi(\theta)_{N+\Delta} = \sum_{a,b \in \{0,1\}} \beta^{(a,b)} \varphi(\theta^{(a,b)}, \mathbf{x})_{N+\Delta}, \quad \beta^{(a,b)} \in \mathbb{C} \quad (3.11)$$

*Proof.* The grid functions  $\varphi(\theta, \mathbf{x})_{N+\Delta}$  for the harmonic frequencies of  $\theta$  are related as:

$$\varphi(\theta^{(a,b)}, \mathbf{x})_{k_1+\Delta_1, k_2+\Delta_2} = \varphi(\theta^{(0,0)}, \mathbf{x})_{k_1+\Delta_1, k_2+\Delta_2} (-1)^{ak_1+bk_2} \gamma^{(a,b)}, \quad (3.12)$$

with  $\gamma^{(a,b)} = e^{i\pi(a\Delta_1+b\Delta_2)}$ . It suffices to note that (3.12) and (3.11) give rise to the following linear system:

$$\begin{bmatrix} \alpha_{00} \\ \alpha_{10} \\ \alpha_{01} \\ \alpha_{11} \end{bmatrix} = \begin{bmatrix} \gamma^{(0,0)} & \gamma^{(1,0)} & \gamma^{(0,1)} & \gamma^{(1,1)} \\ \gamma^{(0,0)} & -\gamma^{(1,0)} & \gamma^{(0,1)} & -\gamma^{(1,1)} \\ \gamma^{(0,0)} & \gamma^{(1,0)} & -\gamma^{(0,1)} & -\gamma^{(1,1)} \\ \gamma^{(0,0)} & -\gamma^{(1,0)} & -\gamma^{(0,1)} & \gamma^{(1,1)} \end{bmatrix} \begin{bmatrix} \beta^{(0,0)} \\ \beta^{(1,0)} \\ \beta^{(0,1)} \\ \beta^{(1,1)} \end{bmatrix}. \quad (3.13)$$

Let  $\Gamma_{\Delta_1, \Delta_2}(\theta)$  denote the system matrix of (3.13). Its regularity follows immediately from the fact that  $|\gamma^{(a,b)}| = 1$  and from the factorization

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \begin{bmatrix} \gamma^{(0,0)} & 0 & 0 & 0 \\ 0 & \gamma^{(1,0)} & 0 & 0 \\ 0 & 0 & \gamma^{(0,1)} & 0 \\ 0 & 0 & 0 & \gamma^{(1,1)} \end{bmatrix}.$$

□

Note that the classical definition (e.g. 4.4.2. in [13]) for the harmonics of a frequency

$$\theta^{(a,b)} = (\theta_1^{(a)}, \theta_2^{(b)}), \quad a, b \in \{0,1\} \quad \text{with } \theta_i^{(c)} := \begin{cases} \theta_i + c\pi & \text{if } \theta_i < 0 \\ \theta_i - c\pi & \text{if } \theta_i \geq 0 \end{cases} \quad (3.14)$$

would require a separate treatment of the negative and positive frequencies.

Lemma 3.3 can be applied separately to the horizontal and vertical components of (3.10). A grid function satisfying (3.10) can be transformed into a linear combination of vectors of the form (3.3)

$$\psi(\theta)_E = \begin{bmatrix} \Phi(\theta^{(0,0)}) & \Phi(\theta^{(1,0)}) & \Phi(\theta^{(0,1)}) & \Phi(\theta^{(1,1)}) \end{bmatrix} \begin{bmatrix} \beta^{(0,0)} \\ \beta^{(1,0)} \\ \beta^{(0,1)} \\ \beta^{(1,1)} \end{bmatrix}, \quad (3.15)$$

with  $\beta^{(a,b)} = [\beta_1^{(a,b)} \ \beta_2^{(a,b)}]^T \in \mathbb{C}^2$  given by

$$\begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \end{bmatrix} = \Gamma_{\frac{1}{2},0}(\theta) \begin{bmatrix} \beta_1^{(0,0)} \\ \beta_1^{(1,0)} \\ \beta_1^{(0,1)} \\ \beta_1^{(1,1)} \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} \alpha_5 \\ \alpha_6 \\ \alpha_7 \\ \alpha_8 \end{bmatrix} = \Gamma_{0,\frac{1}{2}}(\theta) \begin{bmatrix} \beta_2^{(0,0)} \\ \beta_2^{(1,0)} \\ \beta_2^{(0,1)} \\ \beta_2^{(1,1)} \end{bmatrix}. \quad (3.16)$$

Formula (3.15) shows that  $\psi(\theta)_E$  belongs to

$$F_8(\theta) := \bigcup_{a,b \in \{0,1\}} F(\theta^{(a,b)}). \quad (3.17)$$

The combined transformation matrix will be denoted  $T_8(\theta) \in \mathbb{C}^{8 \times 8}$ , with  $0_4$  denoting the zero matrix of size  $4 \times 4$ :

$$T_8(\theta) := \begin{bmatrix} \Gamma_{\frac{1}{2},0} & 0_4 \\ 0_4 & \Gamma_{0,\frac{1}{2}} \end{bmatrix}. \quad (3.18)$$

**4. The Helmholtz splitting in Fourier space.** It is well known (cfr. [7, 9]) that the Helmholtz splitting (1.3) plays a fundamental role in the analysis of solvers for the curl-curl equation. For the two-dimensional rectangular grid studied in this paper (see Figure 2.1), the discrete gradient and curl operators  $G$  and  $C$  are defined as (see [4]):

$$\forall e_{ij} : G(e_{ij}, k) := \begin{cases} -1 & \text{if } n_k = n_i \\ 1 & \text{if } n_k = n_j \\ 0 & \text{else} \end{cases}$$

$$\forall f_{ijkl} : C(f_{ijkl}, e_{rs}) := \begin{cases} 1 & \text{if } e_{rs} = e_{ij}, e_{jk}, e_{kl} \text{ or } e_{li} \\ -1 & \text{if } e_{rs} = e_{ji}, e_{kj}, e_{lk} \text{ or } e_{il} \\ 0 & \text{else} \end{cases}$$

Here,  $e_{ij}$  denotes the oriented edge with start node  $n_i$  and end node  $n_j$ , and  $f_{ijkl}$  the oriented face with boundary formed by the edge sequence  $e_{ij} \rightarrow e_{jk} \rightarrow e_{kl} \rightarrow e_{li}$ . The following Theorem shows that the spaces  $F_e(\theta)$  can also be split as in (1.3).

**THEOREM 4.1** (Fourier-Helmholtz splitting). *For all  $\theta \in \Theta$ ,  $F_e(\theta)$  can be split as*

$$F(\theta) = \phi_g(\theta)_E \oplus \phi_s(\theta)_E, \quad (4.1)$$

with  $\phi_g(\theta)_E \in \text{ran}(G)$  and  $\phi_s(\theta)_E \in \text{ran}(C^T)$ .

*Proof.* If  $\theta \neq (0,0)$ , the vectors  $\varphi(\theta, \mathbf{x})_N$  and  $\varphi(\theta, \mathbf{x})_F$  of the evaluations of  $\varphi(\theta, \mathbf{x})$  in the grid nodes  $N$  and the center nodes  $F$  of the faces can be used to construct vectors in  $F_e(\theta)$  belonging to  $\text{ran}(G)$  and  $\text{ran}(C^T)$  respectively. The subscripts  $\cdot_g$  and  $\cdot_s$  will be used to refer to the gradient and solenoidal vector respectively.

$$\begin{aligned} \phi_g(\theta)_{\mathbf{k}} &:= (G\varphi(\theta, \mathbf{x})_N)_{\mathbf{k}} = \begin{cases} \varphi(\theta, \mathbf{x})_{k_1+\frac{1}{2}, k_2} - \varphi(\theta, \mathbf{x})_{k_1-\frac{1}{2}, k_2} & \text{if } \mathbf{k} \in E_1 \\ \varphi(\theta, \mathbf{x})_{k_1, k_2+\frac{1}{2}} - \varphi(\theta, \mathbf{x})_{k_1, k_2-\frac{1}{2}} & \text{if } \mathbf{k} \in E_2 \end{cases} \quad (4.2) \\ &= \begin{cases} 2i \sin(\frac{\theta_1}{2}) \varphi(\theta, \mathbf{x})_{\mathbf{k}} & \text{if } \mathbf{k} \in E_1 \\ 2i \sin(\frac{\theta_2}{2}) \varphi(\theta, \mathbf{x})_{\mathbf{k}} & \text{if } \mathbf{k} \in E_2 \end{cases} \end{aligned}$$

$$\begin{aligned} \phi_s(\theta)_{\mathbf{k}} &:= (C^T \varphi(\theta, \mathbf{x})_F)_{\mathbf{k}} = \begin{cases} \varphi(\theta, \mathbf{x})_{k_1, k_2+\frac{1}{2}} - \varphi(\theta, \mathbf{x})_{k_1, k_2-\frac{1}{2}} & \text{if } \mathbf{k} \in E_1 \\ \varphi(\theta, \mathbf{x})_{k_1-\frac{1}{2}, k_2} - \varphi(\theta, \mathbf{x})_{k_1+\frac{1}{2}, k_2} & \text{if } \mathbf{k} \in E_2 \end{cases} \quad (4.3) \\ &= \begin{cases} 2i \sin(\frac{\theta_2}{2}) \varphi(\theta, \mathbf{x})_{\mathbf{k}} & \text{if } \mathbf{k} \in E_1 \\ -2i \sin(\frac{\theta_1}{2}) \varphi(\theta, \mathbf{x})_{\mathbf{k}} & \text{if } \mathbf{k} \in E_2. \end{cases} \end{aligned}$$

Since  $\phi_g(\theta)_E$  and  $\phi_s(\theta)_E$  are of the form (3.3) for all  $\theta \in \Theta$ , they belong to  $F_e(\theta)$ . By construction, they belong to  $\text{ran}(G)$  and  $\text{ran}(C^T)$  respectively. Since they are different from the 0-vector for all  $\theta \neq (0, 0)$  and  $\text{ran}(G) \perp \text{ran}(C^T)$ , they are linearly independent. This allows to conclude that the columns of

$$\Phi_H(\theta) := \begin{bmatrix} \phi_g(\theta)_E & \phi_s(\theta)_E \end{bmatrix} \quad (4.4)$$

constitute a basis for  $F_e(\theta)$  for all  $\theta \neq (0, 0)$ .

The above procedure cannot be used for  $\theta = (0, 0)$ , since it would yield  $\phi_g = \phi_s \equiv 0_E$ . In this case, the grid functions  $\psi(\mathbf{x})_N$  and  $\psi(\mathbf{x})_F$ , with  $\psi(h(k_1, k_2)) := 2i(k_1 + k_2)$  can be used instead, which yield:

$$\begin{aligned} \phi_g(\mathbf{0})_{\mathbf{k}} &:= (G\psi(\mathbf{x})_N)_{\mathbf{k}} = \begin{cases} 2i = 2i\varphi(\mathbf{0}, \mathbf{x})_{\mathbf{k}} & \text{if } \mathbf{k} \in E_1 \\ 2i = 2i\varphi(\mathbf{0}, \mathbf{x})_{\mathbf{k}} & \text{if } \mathbf{k} \in E_2 \end{cases} \\ \phi_s(\mathbf{0})_{\mathbf{k}} &:= (C^T\psi(\mathbf{x})_F)_{\mathbf{k}} = \begin{cases} 2i = 2i\varphi(\mathbf{0}, \mathbf{x})_{\mathbf{k}} & \text{if } \mathbf{k} \in E_1 \\ -2i = -2i\varphi(\mathbf{0}, \mathbf{x})_{\mathbf{k}} & \text{if } \mathbf{k} \in E_2. \end{cases} \end{aligned}$$

The conclusion that  $\Phi(\mathbf{0})$  is a basis for  $F_e(\mathbf{0})$  follows.

□

A subscript  $\cdot_H$  will be used to refer to the Fourier-Helmholtz basis (4.4). This basis is related to the decoupled basis  $\Phi(\theta)$  from (3.6) as

$$\Phi_H(\theta) = 2i\Phi(\theta)H(\theta), \quad (4.5)$$

with

$$H(\theta) := \begin{cases} \begin{bmatrix} \sin(\frac{\theta_1}{2}) & \sin(\frac{\theta_2}{2}) \\ \sin(\frac{\theta_2}{2}) & -\sin(\frac{\theta_1}{2}) \end{bmatrix} & \text{if } \theta \neq (0, 0) \\ \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} & \text{if } \theta = (0, 0). \end{cases} \quad (4.6)$$

A representation of a grid function in  $F_e(\theta)$  with respect to the basis  $\Phi_H(\theta)$  can be transformed into a representation with respect to the basis  $\Phi(\theta)$  as follows:

$$\Phi_H(\theta) \begin{bmatrix} \alpha_g \\ \alpha_s \end{bmatrix} = \Phi(\theta) \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix} \iff 2iH(\theta) \begin{bmatrix} \alpha_g \\ \alpha_s \end{bmatrix} = \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix}. \quad (4.7)$$

The LFA presented in this paper will be based on the basis (3.6). For the analysis of the hybrid smoother and for interpretation purposes, however, a transformation to the basis (4.4) will be advantageous.

In the rest of this paper, the frequency argument  $\theta$  will be left out whenever its value is clear from the context. The following abbreviations will be used:

$$\begin{aligned} s_i &= \sin(\frac{\theta_i}{2}) & \tilde{s}_i &= \sin(\theta_i) & s_{i,a} &= \sin(\frac{\theta_i + a\pi}{2}) & \tilde{s}_{i,a} &= \sin(\theta_i + a\pi) \\ c_i &= \cos(\frac{\theta_i}{2}) & \tilde{c}_i &= \cos(\theta_i) & c_{i,a} &= \cos(\frac{\theta_i + a\pi}{2}) & \tilde{c}_{i,a} &= \cos(\theta_i + a\pi). \end{aligned}$$

## 5. Fourier domain representation of the operator.

**5.1. Stiffness part.** Application of (3.8) to the stencils defining  $K_{cc}$  yields:

$$\begin{bmatrix} \beta_1 \\ \beta_2 \end{bmatrix} = \begin{bmatrix} 4s_2^2 & -4s_1s_2 \\ -4s_1s_2 & 4s_1^2 \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix} =: K_{cc}(\theta) \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix}. \quad (5.1)$$

## 2-colour Gauss–Seidel

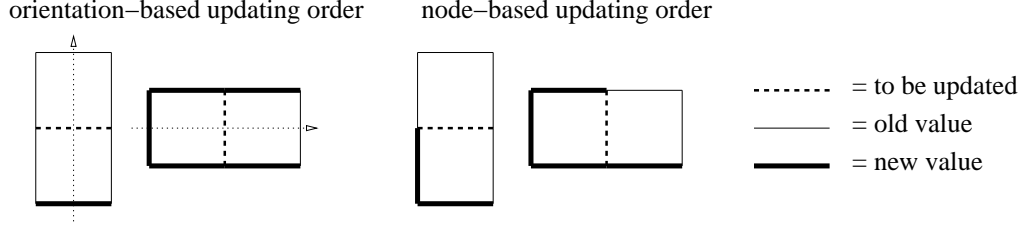


FIG. 6.1. *Orientation-based and node-based updating orders for lexicographic Gauss-Seidel iterations. Each orientation-based iteration consists of 2 subiterations, one for the horizontal and one for the vertical edges. In the node-based iteration, a vertical edge is updated immediately after the horizontal edge with the same start node.*

Using (4.7), (5.1) can be transformed into:

$$\begin{aligned} \begin{bmatrix} \beta_g \\ \beta_s \end{bmatrix} &= H(\theta)^{-1} K_{cc}(\theta) H(\theta) \begin{bmatrix} \alpha_g \\ \alpha_s \end{bmatrix} \\ &= \begin{bmatrix} 0 & 0 \\ 0 & 4(s_1^2 + s_2^2) \end{bmatrix} \begin{bmatrix} \alpha_g \\ \alpha_s \end{bmatrix} =: K_{cc,H}(\theta) \begin{bmatrix} \alpha_g \\ \alpha_s \end{bmatrix}. \end{aligned} \quad (5.2)$$

The nonzero structure of  $K_{cc,H}(\theta)$  allows to identify  $\phi_g(\theta)_E$  and  $\phi_s(\theta)_E$  as the eigenvectors of the discrete curl-curl operator. The zero eigenvalue for  $\phi_g(\theta)_E$  corresponds to the fact that  $\text{ran}(G)$  is the kernel of the discrete curl-curl operator (see [9]). The eigenvalue of  $\phi_s(\theta)_E$  can be interpreted as the Fourier domain counterpart of the equivalence of the vector Laplace operator and the curl-curl operator in the divergence-free part of the Helmholtz splitting. In continuous space, this equivalence follows from the vector identity:

$$\text{curl}(\text{curl}(\vec{A})) = \begin{pmatrix} \text{div}(\text{grad}(A_x)) \\ \text{div}(\text{grad}(A_y)) \\ \text{div}(\text{grad}(A_z)) \end{pmatrix} + \text{grad}(\text{div}(\vec{A})). \quad (5.3)$$

Note, for this purpose, that the eigenvalue for  $\phi_s(\theta)_E$  is identical to the Fourier domain representation of the finite difference discretization  $K_{dg}$  of the div-grad operator on a two-dimensional rectangular grid (see for instance [13]):

$$K_{dg}\varphi(\theta)_N = \frac{4}{h^2}(s_1^2 + s_2^2)\varphi(\theta)_N.$$

**5.2. Mass part.** Application of Theorem 3.1 for the mass part  $\beta_h M$  yields:

$$\beta_h M(\theta) := \frac{\beta_h}{3} \begin{bmatrix} 2 + \tilde{c}_2 & 0 \\ 0 & 2 + \tilde{c}_1 \end{bmatrix} \quad (5.4)$$

Formula (5.4) shows that the columns of  $\Phi(\theta)$  for all  $\theta$  are the eigenvectors of  $M$ .

## 6. Fourier domain representation of the smoother.

**6.1. Classical smoothers.** It can be shown that classical smoothers correspond in the Fourier domain to an iteration based on a matrix splitting of the Fourier domain representation of the operator  $K(\theta) := K_{cc}(\theta) + \beta_h M(\theta) = K^+(\theta) + K^-(\theta)$ :

$$\begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix}^{[\nu+1]} = - (K^+(\theta))^{-1} K^-(\theta) \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix}^{[\nu]} =: S(\theta) \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix}^{[\nu]} \quad (6.1)$$

A Jacobi iteration with damping factor  $d_{cc}$  applied to a linear system of the form  $Kx = b$  can be written as

$$x^{[\nu+1]} = x^{[\nu]} + \frac{1}{d_{cc}} \text{diag}(K)^{-1}(b - Kx^{[\nu]}). \quad (6.2)$$

For  $K = K_{cc} + \beta_h M$ , this corresponds to, in the Fourier domain,

$$K^+(\theta) = \begin{bmatrix} d_{cc}(2 + \frac{2\beta_h}{3}) & 0 \\ 0 & d_{cc}(2 + \frac{2\beta_h}{3}) \end{bmatrix}.$$

For lexicographic Gauss-Seidel iterations, the splitting of  $K(\theta)$  depends on the updating order. We consider two examples. In the orientation-based edge ordering of Figure 6.1, the horizontal edges are updated first. This yields, after some straightforward but technical computations, a matrix splitting with

$$K^+(\theta) = \begin{bmatrix} (2 + \frac{2\beta_h}{3}) - (1 - \frac{\beta_h}{6})\varphi_{0,-1} & 0 \\ -4s_1s_2 & (2 + \frac{2\beta_h}{3}) - (1 - \frac{\beta_h}{6})\varphi_{-1,0} \end{bmatrix}.$$

Here,  $\varphi_{k_1, k_2}$  is used as a shorthand for  $\varphi(\theta, \mathbf{x})_{\mathbf{k}}$ , with  $\mathbf{k} = (k_1, k_2)$ . In the node-based edge ordering of Figure 6.1, edges with the same start node are updated consecutively. This yields a matrix splitting with:

$$K^+(\theta) = \begin{bmatrix} (2 + \frac{2\beta_h}{3}) - (1 - \frac{\beta_h}{6})\varphi_{0,-1} & \varphi_{-\frac{1}{2}, -\frac{1}{2}} \\ \varphi_{-\frac{1}{2}, -\frac{1}{2}} - \varphi_{-\frac{1}{2}, \frac{1}{2}} - \varphi_{\frac{1}{2}, -\frac{1}{2}} & (2 + \frac{2\beta_h}{3}) - (1 - \frac{\beta_h}{6})\varphi_{-1,0} \end{bmatrix}.$$

Formula (6.1) shows that the properties of smoothers of this type can be studied for each frequency  $\theta$  separately by means of the spectral radius of  $S(\theta)$

$$\rho(S(\theta)) := \max\{|\lambda| : \lambda \in \sigma(S(\theta))\}. \quad (6.3)$$

The left picture of Figure 6.2 shows  $\rho(S(\theta))$  as a function of  $\theta$  for node-based Gauss-Seidel iterations applied to (1.2). Since in a multigrid context, it is the task of a smoother to reduce the high frequency error components, the smoothing properties of this smoother are very poor. This is due to the non-ellipticity of the curl-curl equation in the gradient part  $\text{ran}(G)$  of the Helmholtz splitting (1.3). We refer the reader to [7] for a further analysis of this phenomenon.

*Remark.* With respect to the Fourier-Helmholtz basis (4.4), the Fourier domain representation of damped Jacobi iterations with  $\beta = 0$  and damping factor  $d_{cc}$  reads

$$\begin{bmatrix} \alpha_g \\ \alpha_s \end{bmatrix}^{[\nu+1]} = \begin{bmatrix} 1 & 0 \\ 0 & 1 - \frac{2}{d_{cc}}(s_1^2 + s_2^2) \end{bmatrix} \begin{bmatrix} \alpha_g \\ \alpha_s \end{bmatrix}^{[\nu]}. \quad (6.4)$$

This illustrates the well known fact that undamped Jacobi ( $d_{cc} = 1$ ) diverges for the curl-curl equation. The damped iteration will converge for any  $d_{cc} \geq 2$ . This also reveals that in divergence-free space, this smoother behaves in the same way as damped Jacobi iterations for the finite difference discretization of the Laplace equation  $\text{div}(\text{grad}(V)) = \rho$  with damping factor  $d_{dg} = d_{cc}/2$ . Indeed, in the Fourier domain, the error equation for the latter satisfies (see [13])

$$\alpha^{[\nu+1]} = \left(1 - \frac{1}{d_{dg}}(s_1^2 + s_2^2)\right) \alpha^{[\nu]}.$$

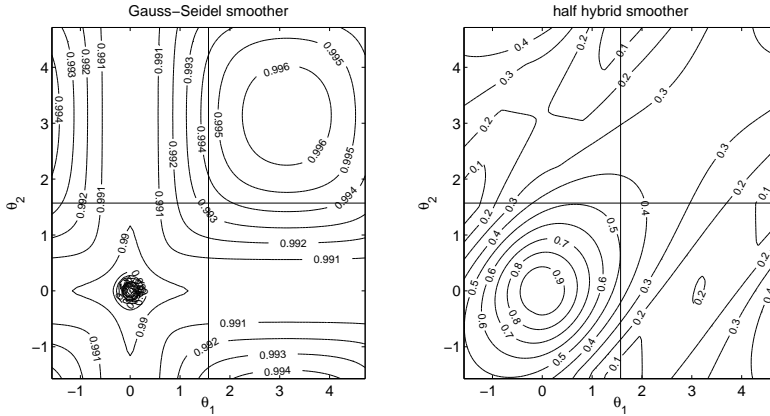


FIG. 6.2. Contourplot of  $\rho(S(\theta))$  as a function of  $\theta$  with  $\beta_h = 0.01$  for node-based Gauss-Seidel iterations (left) and for the hybrid smoother consisting of 1 iteration of node-based Gauss-Seidel type in  $\mathcal{E}$  followed by 1 iteration of lexicographic Gauss-Seidel type in  $\mathcal{N}$  (right). Contrary to the hybrid smoother, Gauss-Seidel iterations don't achieve satisfactory smoothing of the high frequency errors.

**6.2. Hybrid smoother.** The poor smoothing properties of classical smoothers can be improved by an additional smoothing step on an auxiliary nodal diffusion system  $K_n x_n = b_n$  (see [7]). Formally, one iteration of this so-called hybrid smoother is equivalent to a 2-level multigrid cycle using the gradient matrix  $G$  as the prolongation matrix. This cycle is composed of the following 5 steps:

1.  $\epsilon_{pre}$  presmoothing steps in  $\mathcal{E}$  on  $Kx = (K_{cc} + \beta_h M)x = b$
2. restriction to  $\mathcal{N}$ :  $b_n = G^T(b - Kx)$
3. smoothing in  $\mathcal{N}$  on  $K_n x_n = b_n$  with zero initial guess
4. prolongation to  $\mathcal{E}$ :  $x \leftarrow x + Gx_n$
5.  $\epsilon_{post}$  postsmoothing steps in  $\mathcal{E}$  on  $Kx = b$

Since the Helmholtz splitting (1.3) is explicitly taken into account in the hybrid smoother via the discrete gradient matrix  $G$ , the Fourier-Helmholtz basis (4.4) will be particularly helpful in the LFA. First, the nodal part of the hybrid smoother, consisting of steps 2, 3 and 4, is studied separately. We start with the left multiplication by  $G^T$  in step 2. The application of  $G^T$  to a grid function of the form (3.3) yields, with  $\mathbf{k} = (k_1, k_2) \in N$ :

$$\begin{aligned} \left( G^T \Phi(\theta) \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix} \right)_{\mathbf{k}} &= -\alpha_1 \left( \varphi(\theta)_{k_1 + \frac{1}{2}, k_2} - \varphi(\theta)_{k_1 - \frac{1}{2}, k_2} \right) \\ &\quad - \alpha_2 \left( \varphi(\theta)_{k_1, k_2 + \frac{1}{2}} - \varphi(\theta)_{k_1, k_2 - \frac{1}{2}} \right) \\ &= \beta \varphi(\theta)_{\mathbf{k}}, \end{aligned} \quad (6.5)$$

with

$$\beta = -2i \begin{bmatrix} s_1 & s_2 \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix} =: G^T(\theta) \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix}. \quad (6.6)$$

Using (4.7), the relation (6.6) can be transformed into:

$$\beta = \begin{bmatrix} 4(s_1^2 + s_2^2) & 0 \end{bmatrix} \begin{bmatrix} \alpha_g \\ \alpha_s \end{bmatrix} =: G_H^T(\theta) \begin{bmatrix} \alpha_g \\ \alpha_s \end{bmatrix}. \quad (6.7)$$

Formula (6.7) is the Fourier domain counterpart of the transpose of the topological identity  $CG \equiv 0$ , as it shows that the restriction of vectors belonging to  $\text{ran}(C^T)$  by  $G^T$  is 0. Now, the restriction of the edge residual  $y = G^T(b - Kx)$  in step 2 can be represented in the Fourier domain. With  $K_H(\theta)$ ,  $K_{cc,H}(\theta)$  and  $M_H(\theta)$  the Fourier symbols of  $K$ ,  $K_{cc}$  and  $M$  with respect to the Helmholtz-Fourier basis (4.4), we obtain:

$$G_H^T(\theta)K_H(\theta) \begin{bmatrix} \alpha_g \\ \alpha_s \end{bmatrix} = G_H^T(\theta) (K_{cc,H}(\theta) + \beta_h M_H(\theta)) \begin{bmatrix} \alpha_g \\ \alpha_s \end{bmatrix} \quad (6.8)$$

$$= \beta_h G_H^T(\theta) M_H(\theta) \begin{bmatrix} \alpha_g \\ \alpha_s \end{bmatrix} \quad (6.9)$$

$$= \frac{\beta_h}{3} \begin{bmatrix} 4(s_1^2 + s_2^2) & 0 \end{bmatrix} H^{-1}(\theta) \begin{bmatrix} 2 + \tilde{c}_2 & 0 \\ 0 & 2 + \tilde{c}_1 \end{bmatrix} H(\theta) \begin{bmatrix} \alpha_g \\ \alpha_s \end{bmatrix} \\ = \frac{4\beta_h}{3} \begin{bmatrix} (2 + \tilde{c}_2)s_1^2 + (2 + \tilde{c}_1)s_2^2 & s_1 s_2 (\tilde{c}_2 - \tilde{c}_1) \end{bmatrix} \begin{bmatrix} \alpha_g \\ \alpha_s \end{bmatrix}.$$

For  $\theta \neq (0, 0)$ , the prolongation by  $G$  of  $\varphi(\theta)_N \alpha$ , with  $\alpha \in \mathbb{C}$ , in step 4 yields a scaled gradient basis function by (4.2). For  $\theta = (0, 0)$ , we refer to the remark at the end of this section. In the Fourier domain and using (4.4), step 4 reads:

$$(G\varphi(\theta)_N \alpha)_E = \Phi_H(\theta) \begin{bmatrix} \beta_g \\ \beta_s \end{bmatrix}, \quad \text{with} \quad \begin{bmatrix} \beta_g \\ \beta_s \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \alpha \quad (6.10)$$

The auxiliary nodal matrix  $K_n$  for step 3 is usually built as the Galerkin product  $G^T K G$ . This corresponds to the stencil for the finite element discretization using bilinear finite elements of the Poisson equation on the same grid, scaled by  $\beta_h$ :

$$\beta_h \begin{bmatrix} -1/3 & -1/3 & -1/3 \\ -1/3 & 8/3 & -1/3 \\ -1/3 & -1/3 & -1/3 \end{bmatrix}, \quad (6.11)$$

and to the following Fourier domain representation:

$$K_n(\theta) = G_H^T(\theta)K_H(\theta)G_H(\theta) = \beta_h G_H^T(\theta)M_H(\theta)G_H(\theta) \quad (6.12) \\ = \frac{\beta_h}{3} \begin{bmatrix} 4(s_1^2 + s_2^2) & 0 \end{bmatrix} H^{-1}(\theta) \begin{bmatrix} 2 + \tilde{c}_2 & 0 \\ 0 & 2 + \tilde{c}_1 \end{bmatrix} H(\theta) \begin{bmatrix} 1 \\ 0 \end{bmatrix} \\ = \frac{4\beta_h}{3} ((2 + \tilde{c}_2)s_1^2 + (2 + \tilde{c}_1)s_2^2).$$

Now, the combined effect of steps 2, 3 and 4 can be studied. Suppose the smoother applied to the auxiliary system  $K_n x_n = b_n$  is a linear operator  $S_n$  with eigenvectors  $\varphi(\theta)_N$  and corresponding eigenvalues  $S_n(\theta)$ . This is the case if classical smoothers, such as Jacobi or lexicographic Gauss-Seidel iterations, are used (see [13]). Let  $x_n^*$  denote the exact solution of the auxiliary system, i.e.  $x_n^* = K_n^{-1} b_n = K_n^{-1} G^T (b - Kx)$ , and let  $x_n^{[1]}$  and  $e_n^{[1]} = x_n^* - x_n^{[1]}$  be an approximate solution and the corresponding error after the smoothing step 3. Since this smoothing starts from a zero initial guess  $x_n^{[0]} = 0$ , the initial error  $e_n^{[0]}$  is identical to  $x_n^*$ . Now,  $x_n^{[1]}$  can be reconstructed from  $e_n^{[1]} = x_n^* - x_n^{[1]} = S_n e_n^{[0]} = S_n x_n^*$ , which gives

$$x_n^{[1]} = (I - S_n)x_n^* = (I - S_n)K_n^{-1}G^T(b - Kx)$$

Prolongation of  $x_n^{[1]}$  yields, with  $e = x^* - x$  denoting the error of (1.2):

$$x \leftarrow x + Gx_n^{[1]} \iff e \leftarrow (I - G(I - S_n)K_n^{-1}G^TK)e$$

In frequency space, this becomes:

$$\begin{aligned} \begin{bmatrix} \alpha_g \\ \alpha_s \end{bmatrix} &\leftarrow (I - G_H(\theta)(1 - S_n(\theta))K_n^{-1}(\theta)G_H^T(\theta)K_H(\theta)) \begin{bmatrix} \alpha_g \\ \alpha_s \end{bmatrix} \\ &\leftarrow \left( I - \begin{bmatrix} 1 \\ 0 \end{bmatrix} (1 - S_n(\theta)) \begin{bmatrix} 1 & \frac{s_1 s_2 (\tilde{c}_2 - \tilde{c}_1)}{(2 + \tilde{c}_2)s_1^2 + (2 + \tilde{c}_1)s_2^2} \end{bmatrix} \right) \begin{bmatrix} \alpha_g \\ \alpha_s \end{bmatrix} \\ &\leftarrow \begin{bmatrix} S_n(\theta) & \frac{(S_n(\theta) - 1)s_1 s_2 (\tilde{c}_2 - \tilde{c}_1)}{(2 + \tilde{c}_2)s_1^2 + (2 + \tilde{c}_1)s_2^2} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \alpha_g \\ \alpha_s \end{bmatrix} := S_{h234}(\theta) \begin{bmatrix} \alpha_g \\ \alpha_s \end{bmatrix} \end{aligned} \quad (6.13)$$

Formula (6.13) shows that the nodal part of the hybrid smoother leaves the solenoidal errors untouched (lower right entry). If no solenoidal errors are present, the gradient errors are reduced in the same way as the error of the auxiliary system (upper left entry). The upper right entry represents the aliasing contribution of solenoidal errors. With  $S_{h1}$  and  $S_{h5}$  denoting the Fourier domain representation with respect to the Fourier-Helmholtz basis (4.4) of the smoothers in steps 1 and 5, the frequency response of the hybrid smoother reads:

$$S_h(\theta) \begin{bmatrix} \alpha_g \\ \alpha_s \end{bmatrix} := S_{h5}^{\epsilon_{post}}(\theta) \cdot S_{h234}(\theta) \cdot S_{h1}^{\epsilon_{pre}}(\theta) \cdot \begin{bmatrix} \alpha_g \\ \alpha_s \end{bmatrix}. \quad (6.14)$$

Formula (6.14) shows that, similarly as for classical smoothers, the properties of the hybrid smoother can be studied for each frequency separately. In Figure 6.2, the smoothing properties of a classical smoother and of its hybrid extension are compared. The figures show that the hybrid smoother achieves a satisfactory smoothing of the high frequency error components, contrary to its classical counterpart.

*Remark.* The construction of  $K_n$  as the Galerkin product  $G^TKG$  is optimal in the following sense. Let  $x^*$  satisfy  $Kx^* = b$ . If the error  $e = x^* - x = K^{-1}b - x$  belongs to  $\text{ran}(G)$ , i.e.,  $e = Gp$ , then the exact solution of the auxiliary system  $K_n x_n = G^T(b - Kx)$  is equal to  $p$ . This follows from

$$K_n p = G^T K G p = G^T (b - Kx).$$

Hence, if the auxiliary system is solved exactly, the error of the original system will be totally eliminated.

*Remark.* Different configurations of the hybrid smoother are possible. They depend on the number of pre- and post-smoothing steps for the edge smoother in steps 1 and 5 and on the configuration of the nodal smoother in step 3. For classical smoothers of Jacobi or Gauss-Seidel type, the configuration of the nodal smoother can be summarized as:

$$\text{nodal smoother} = \begin{cases} 1. & \nu_{pre} \text{ iterations in presmoothing mode} \\ 2. & \nu_{post} \text{ iterations in postsmoothing mode.} \end{cases}$$

Both smoothing modes for the edge and nodal smoother differ only in the order in which the unknowns are updated: the updating order in postsmoothing mode is the reverse of the updating order in presmoothing mode. Popular configurations for the hybrid smoother are the so-called half and full hybrid smoother (see for instance [11]). The full hybrid smoother has  $\epsilon_{pre} = \epsilon_{post} = \nu_{pre} = \nu_{post} = 1$ . The half hybrid smoother has  $\epsilon_{pre} = \nu_{pre} = 1$  and  $\epsilon_{post} = \nu_{post} = 0$  in presmoothing mode, and  $\epsilon_{pre} = \nu_{pre} = 0$  and  $\epsilon_{post} = \nu_{post} = 1$  in postsmoothing mode.

*Remark.* The above derivation does not hold for  $\theta = (0, 0)$ , since  $K_n((0, 0)) = 0$  and  $G((0, 0)) = [0 \ 0]^T$ , which would give rise to a zero-by-zero division. This singularity of  $K_n$  is a consequence of the absence of boundary conditions, which is generic for a *local* Fourier analysis (cfr. [13], p.109). The appropriate introduction of boundary conditions in the auxiliary nodal system can be studied in a *global* Fourier analysis. For the results presented in this paper, the nodal part of the hybrid smoother is not executed for  $\theta = (0, 0)$ . This is motivated by the observation that the nodal part of the hybrid smoother only affects errors in  $\text{ran}(G)$ , which belong to the kernel of the system matrix  $K$  for  $\theta = (0, 0)$ .

**6.3. Overlapping block smoother.** In [1], it is proved that good smoothing properties can be achieved by using a certain overlapping block smoother. One iteration of the so-called AFW-smoother proposed in [1] consists of a sequence of local solves, one corresponding to each node, in which all edges connected to that node are updated simultaneously. Using MATLAB-notation for extracting submatrices and with  $e_n$  denoting the set of the edges containing node  $n$ , this algorithm consists of a sequence of local block solves for all nodes  $n$ :

$$x(e_n) = x(e_n) + K(e_n, e_n)^{-1} (b(e_n) - K(e_n, \cdot)x).$$

The AFW-smoother can be analyzed in  $F_8(\theta)$  using the 8-color parameterization (3.10). The Fourier domain representation in  $F_8(\theta)$  of the system matrix can be built directly by using the 8-color variant of formula (3.8) or by applying the similarity transformation (3.18) to the block diagonal matrix representing the operator for the harmonic frequencies of  $\theta$ :

$$K_8(\theta) = T_8(\theta) \begin{bmatrix} K(\theta^{(0,0)}) & 0 & 0 & 0 \\ 0 & K(\theta^{(1,0)}) & 0 & 0 \\ 0 & 0 & K(\theta^{(0,1)}) & 0 \\ 0 & 0 & 0 & K(\theta^{(1,1)}) \end{bmatrix} T_8^{-1}(\theta).$$

This setting allows to analyze the AFW-smoother, if it consists of the following four steps, which can be executed in any order:

1. updating simulatenously all edges 3 – 4 – 6 – 8
2. updating simulatenously all edges 1 – 2 – 5 – 7
3. updating simulatenously all edges 1 – 2 – 6 – 8
4. updating simulatenously all edges 3 – 4 – 5 – 7.

The update order is illustrated in Figure 6.3. The error equation for the step in which the colors  $c = [i, j, k, l]$  are updated, corresponds to a local block solve in the Fourier domain. With  $\alpha = [\alpha_1 \dots \alpha_8]^T$ , this local block solve reads, using MATLAB notation,

$$\alpha(c) \leftarrow - (K_8(c, c))^{-1} K_8(c, \bar{c})\alpha(\bar{c}).$$

Here  $\bar{c}$  denotes the set of colors that are not updated. The Fourier domain iteration matrix  $S_c$  for this step reads, with  $I_4$  denoting the unit matrix of dimension 4:

$$S_c([c \ \bar{c}], [c \ \bar{c}]) = \begin{bmatrix} 0 & - (K_8(c, c))^{-1} K_8(c, \bar{c}) \\ 0 & I_4 \end{bmatrix}.$$

This allows the global Fourier domain iteration matrix of the AFW-smoother with respect to the parameterization (3.15) to be constructed as:

$$S_{AFW}(\theta) = T_8(\theta)^{-1} \cdot S_{3457}(\theta) \cdot S_{1268}(\theta) \cdot S_{1257}(\theta) \cdot S_{3468}(\theta) \cdot T_8(\theta).$$

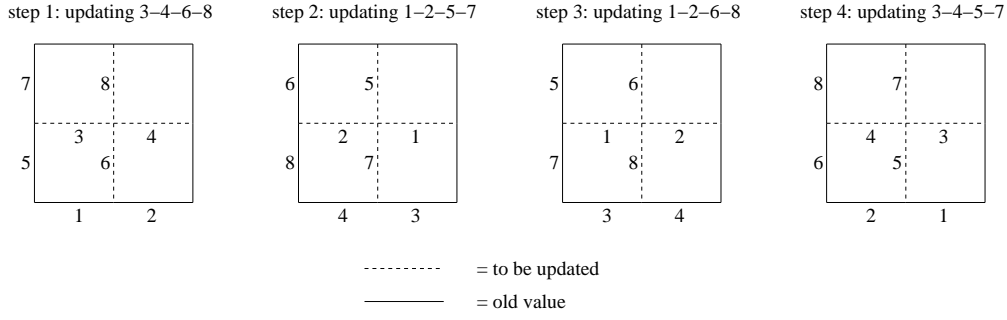


FIG. 6.3. Updating order for the different steps of the AFW-smoother analyzed in the LFA. Each of the pictures zooms in on a different part of the grid, showing that each edge to be updated in a step is connected to exactly 3 other edges to be updated in the same step.

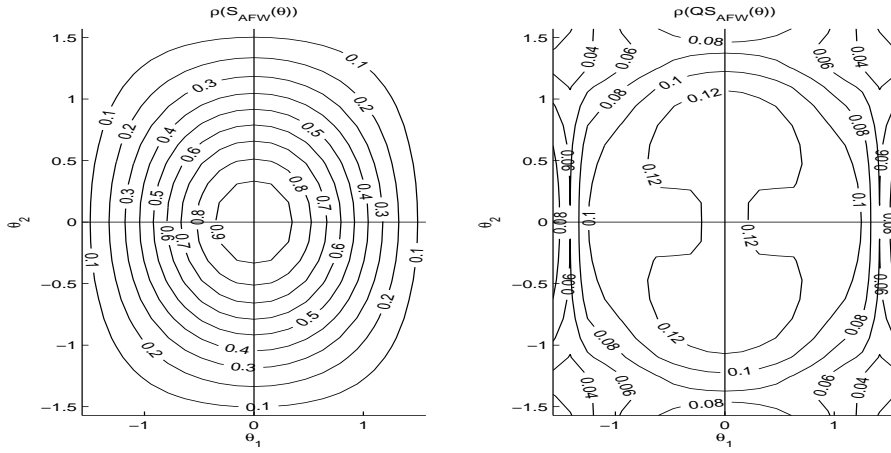


FIG. 6.4. Contour plot of  $\rho(S_{AFW}(\theta))$  and  $\rho(CS_{AFW}(\theta))$  for  $\beta = 0.01$ .

To study the smoothing properties, the high frequencies need to be isolated from the low frequencies. This is not possible here, since in general,  $S_{AFW}(\theta)$  is dense. However, an ideal coarse grid operator  $C$

$$C := \begin{bmatrix} 0_2 & 0_2 & 0_2 & 0_2 \\ 0_2 & I_2 & 0_2 & 0_2 \\ 0_2 & 0_2 & I_2 & 0_2 \\ 0_2 & 0_2 & 0_2 & I_2 \end{bmatrix}, \quad \text{with } 0_2 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}, \quad I_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

can be used to suppress the low frequency components (cfr. [13]). The spectral radius of  $CS_{AFW}(\theta)$  is a measure for the smoothing properties of the AFW-smoother in  $F_8(\theta)$  (see Figure 6.4).

## 7. Fourier domain representation of the intergrid transfer operators.

**7.1. Coarse grid.** We consider the coarse grid with mesh size  $\tilde{h} = 2h$

$$G_{\tilde{h}} := \{\mathbf{x} = \mathbf{k}\tilde{h} \mid \mathbf{k} \in \mathbb{Z}^2\}. \quad (7.1)$$

In the rest of this paper, a superscript  $\tilde{\cdot}$  will be used to refer to the coarse grid. Similarly as (3.1), we define

$$\tilde{\varphi}(\theta, \mathbf{x}) := e^{i\theta \cdot \mathbf{x}/\tilde{h}} \quad \text{with } \theta, \mathbf{x} \in \mathbb{R}^2. \quad (7.2)$$

To distinguish the evaluation operators (2.2) with respect to the fine and coarse grid, the grid size will be indicated in superscript as

$$\overset{h}{\mathbf{k}} : f(\mathbf{x}) \mapsto f(\mathbf{x})_{\mathbf{k}}^h := f(hk_1, hk_2). \quad (7.3)$$

The fine and coarse evaluation operators and the grid functions (3.1) and (7.2) are related as follows, with  $\mathbf{r} \in \mathbb{R}^2$ :

$$\varphi(\theta, \mathbf{x})_{\mathbf{r}}^h = \tilde{\varphi}(2\theta, \mathbf{x})_{\mathbf{r}}^h = \tilde{\varphi}(\theta, \mathbf{x})_{\mathbf{r}}^{\tilde{h}} = \tilde{\varphi}(\theta, \mathbf{x})_{2\mathbf{r}}^h. \quad (7.4)$$

Due to the displacement  $\mathbf{\Delta} = (\Delta_1, \Delta_2)$  of the evaluation points, the aliasing of the fine Fourier modes on the coarse grid is more complicated than in LFA for node-based problems, and reads, with  $\theta \in \Theta^{\text{low}}$ ,  $\mathbf{k} = (k_1, k_2) \in N$  and  $a, b \in \{0, 1\}$ :

$$\begin{aligned} \varphi(\theta^{(a,b)}, \mathbf{x})_{\mathbf{k}+\mathbf{\Delta}}^{\tilde{h}} &= e^{i((\theta_1+a\pi, \theta_2+b\pi)(k_1+\Delta_1, k_2+\Delta_2)\tilde{h}/h)} \\ &= \varphi(\theta^{(0,0)}, \mathbf{x})_{\mathbf{k}+\mathbf{\Delta}}^{\tilde{h}} e^{2\pi i(a\Delta_1+b\Delta_2)} e^{2\pi i(ak_1+bk_2)} \\ &= \tilde{\varphi}(2\theta^{(0,0)}, \mathbf{x})_{\mathbf{k}+\mathbf{\Delta}}^{\tilde{h}} e^{i2\pi(a\Delta_1+b\Delta_2)}. \end{aligned} \quad (7.5)$$

**7.2. Restriction.** We consider the classical Nédélec restriction [7], which reads for the horizontal and vertical edges in stencil notation:

$$\begin{bmatrix} \circ & 1/4 & \circ & 1/4 & \circ \\ 0 & & 0 & & 0 \\ \bullet & 1/2 & \circ & 1/2 & \bullet \\ 0 & & 0 & & 0 \\ \circ & 1/4 & \circ & 1/4 & \circ \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} \circ & 0 & \bullet & 0 & \circ \\ 1/4 & & 1/2 & & 1/4 \\ \circ & 0 & \circ & 0 & \circ \\ 1/4 & & 1/2 & & 1/4 \\ \circ & 0 & \bullet & 0 & \circ \end{bmatrix}. \quad (7.6)$$

Since the horizontal and vertical components are decoupled with respect to the restriction operator (7.6), the decoupled parameterization (3.3) of  $F_e(\theta)$  is a natural choice for the analysis of the restriction. Using (7.4) and (7.5), the restriction of the horizontal components reads:

$$\begin{aligned} \left( R\phi_1(\theta^{(a,b)}) \right)_{k_1+\frac{1}{2}, k_2}^{\tilde{h}} &= \frac{1}{2} \left( \varphi(\theta^{(a,b)})_{2k_1+\frac{1}{2}, 2k_2}^h + \varphi(\theta^{(a,b)})_{2k_1+\frac{3}{2}, 2k_2}^h \right) \\ &\quad + \frac{1}{4} \left( \varphi(\theta^{(a,b)})_{2k_1+\frac{1}{2}, 2k_2+1}^h + \varphi(\theta^{(a,b)})_{2k_1+\frac{3}{2}, 2k_2+1}^h \right. \\ &\quad \left. + \varphi(\theta^{(a,b)})_{2k_1+\frac{1}{2}, 2k_2-1}^h + \varphi(\theta^{(a,b)})_{2k_1+\frac{3}{2}, 2k_2-1}^h \right) \\ &= c_{1,a}(1 + \tilde{c}_{2,b})\varphi(\theta^{(a,b)})_{2k_1+1, 2k_2}^h \\ &= 2c_{1,a}c_{2,b}^2\tilde{\varphi}(2\theta^{(a,b)})_{k_1+\frac{1}{2}, k_2}^{\tilde{h}} \\ &= 2c_{1,a}c_{2,b}^2(-1)^a\tilde{\varphi}(2\theta^{(0,0)})_{k_1+\frac{1}{2}, k_2}^{\tilde{h}}. \end{aligned} \quad (7.7)$$

Similarly, we get for the restriction of the vertical components:

$$\left( R\phi_2(\theta^{(a,b)}) \right)_{k_1, k_2+\frac{1}{2}}^{\tilde{h}} = 2c_{2,b}c_{1,a}^2(-1)^b\tilde{\varphi}(2\theta^{(0,0)})_{k_1, k_2+\frac{1}{2}}^{\tilde{h}}. \quad (7.8)$$

Formulae (7.7) and (7.8) can be combined into the form (3.3):

$$R\Phi(\theta^{(a,b)}) \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix} = \tilde{\Phi}(2\theta^{(0,0)}) \begin{bmatrix} \tilde{\alpha}_1 \\ \tilde{\alpha}_2 \end{bmatrix}, \quad (7.9)$$

with

$$\begin{bmatrix} \tilde{\alpha}_1 \\ \tilde{\alpha}_2 \end{bmatrix} = \begin{bmatrix} 2c_{1,a}c_{2,b}^2(-1)^a & 0 \\ 0 & 2c_{2,b}c_{1,a}^2(-1)^b \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix} := R(\theta^{(a,b)}) \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix} \quad (7.10)$$

and

$$\tilde{\Phi}(\theta) := \begin{bmatrix} \tilde{\varphi}_1(\theta)_{\tilde{E}}^{\tilde{h}} & \tilde{\varphi}_2(\theta)_{\tilde{E}}^{\tilde{h}} \end{bmatrix} := \begin{bmatrix} \tilde{\varphi}(\theta, \mathbf{x})_{E_1}^{\tilde{h}} & 0_{E_1}^{\tilde{h}} \\ 0_{E_2}^{\tilde{h}} & \tilde{\varphi}(\theta, \mathbf{x})_{E_2}^{\tilde{h}} \end{bmatrix}. \quad (7.11)$$

The space spanned by (7.11) is the coarse equivalent of  $F_e(\theta)$ , and will be denoted  $\tilde{F}_e(\theta)$ . Formula (7.9) shows that the spaces  $F_e(\theta^{(a,b)})$  for the harmonic frequencies of  $\theta^{(0,0)} \in \Theta^{\text{low}}$  are mapped to the *same* space  $\tilde{F}_e(2\theta^{(0,0)})$  by the restriction operator. Using the coarse equivalent of (4.5)

$$\tilde{\Phi}_H(\theta) = 2i\tilde{\Phi}(\theta)H(\theta), \quad (7.12)$$

(7.9) and (7.10) can be transformed into:

$$R\Phi_H(\theta^{(a,b)}) \begin{bmatrix} \alpha_g \\ \alpha_s \end{bmatrix} = \tilde{\Phi}_H(2\theta^{(0,0)}) \begin{bmatrix} \tilde{\alpha}_g \\ \tilde{\alpha}_s \end{bmatrix} \quad (7.13)$$

with

$$\begin{aligned} \begin{bmatrix} \tilde{\alpha}_g \\ \tilde{\alpha}_s \end{bmatrix} &= \left( H(2\theta^{(0,0)}) \right)^{-1} R(\theta)H(\theta^{(a,b)}) \begin{bmatrix} \alpha_g \\ \alpha_s \end{bmatrix} \\ &= \frac{1}{\tilde{s}_{1,a}^2 + \tilde{s}_{2,b}^2} \begin{bmatrix} (-1)^a \tilde{s}_{1,a} & (-1)^b \tilde{s}_{2,b} \\ (-1)^b \tilde{s}_{2,b} & -(-1)^a \tilde{s}_{1,a} \end{bmatrix} R(\theta^{(a,b)}) \begin{bmatrix} s_{1,a} & s_{2,b} \\ s_{2,b} & -s_{1,a} \end{bmatrix} \\ &= \begin{bmatrix} \frac{\tilde{s}_{1,a}^2 c_{2,b}^2 + \tilde{s}_{2,b}^2 c_{1,a}^2}{\tilde{s}_{1,a}^2 + \tilde{s}_{2,b}^2} & 0 \\ (-1)^{a+b} \frac{\tilde{s}_{1,a} \tilde{s}_{2,b} (c_{2,b}^2 - c_{1,a}^2)}{\tilde{s}_{1,a}^2 + \tilde{s}_{2,b}^2} & (-1)^{a+b} c_{1,a} c_{2,b} \end{bmatrix} \begin{bmatrix} \alpha_g \\ \alpha_s \end{bmatrix} \\ &:= R_H(\theta^{(a,b)}) \begin{bmatrix} \alpha_g \\ \alpha_s \end{bmatrix}. \end{aligned} \quad (7.14)$$

*Remark.* The nonzero structure of  $R_H(\theta)$  shows that the restriction of a fine solenoidal vector is a coarse solenoidal vector. This property corresponds to the transpose of the edge-face commutation property, which reads, with  $P_f, P_e, C_f$  and  $C_c$  denoting respectively the face and edge prolongators and the fine and coarse discrete curl operators:

$$P_f C_c = C_f P_e \iff C_c^T P_f^T = P_e^T C_f^T.$$

**7.3. Prolongation.** We consider the classical Nédélec prolongation indicated by Figure (7.6). The horizontal and vertical prolongation can be studied separately using

the decoupled basis (7.11). This yields for the horizontal case:

$$\begin{aligned} \left(P\tilde{\phi}_1(2\theta)\right)_{k_1+\frac{1}{2},k_2}^h &= \begin{cases} \frac{1}{2}\varphi(\theta)_{k_1+1,k_2}^h & k_1, k_2 \text{ even} \\ \frac{1}{2}\varphi(\theta)_{k_1,k_2}^h & k_1 \text{ odd}, k_2 \text{ even} \\ \frac{1}{4}\left(\varphi(\theta)_{k_1+1,k_2-1}^h + \varphi(\theta)_{k_1+1,k_2+1}^h\right) & k_1 \text{ even}, k_2 \text{ odd} \\ \frac{1}{4}\left(\varphi(\theta)_{k_1,k_2-1}^h + \varphi(\theta)_{k_1,k_2+1}^h\right) & k_1, k_2 \text{ odd} \end{cases} \\ &= \varphi(\theta)_{k_1+\frac{1}{2},k_2}^h \cdot \begin{cases} \frac{1}{2}\varphi(\theta)_{\frac{1}{2},0}^h & k_1, k_2 \text{ even} \\ \frac{1}{2}\varphi(\theta)_{-\frac{1}{2},0}^h & k_1 \text{ odd}, k_2 \text{ even} \\ \frac{1}{2}\tilde{c}_2\varphi(\theta)_{\frac{1}{2},0}^h & k_1 \text{ even}, k_2 \text{ odd} \\ \frac{1}{2}\tilde{c}_2\varphi(\theta)_{-\frac{1}{2},0}^h & k_1, k_2 \text{ odd} \end{cases} \end{aligned} \quad (7.15)$$

Similarly, we have for the vertical prolongation:

$$\left(P\tilde{\phi}_2(2\theta)\right)_{k_1,k_2+\frac{1}{2}}^h = \varphi(\theta)_{k_1,k_2+\frac{1}{2}} \cdot \begin{cases} \frac{1}{2}\varphi(\theta)_{0,\frac{1}{2}} & \text{if } k_1, k_2 \text{ even} \\ \frac{1}{2}\tilde{c}_1\varphi(\theta)_{0,\frac{1}{2}} & \text{if } k_1 \text{ odd}, k_2 \text{ even} \\ \frac{1}{2}\varphi(\theta)_{0,-\frac{1}{2}} & \text{if } k_1 \text{ even}, k_2 \text{ odd} \\ \frac{1}{2}\tilde{c}_1\varphi(\theta)_{0,-\frac{1}{2}} & \text{if } k_1, k_2 \text{ odd} \end{cases} \quad (7.16)$$

Using (3.16) and Lemma 3.3, (7.15) can be written in the form (3.15), with:

$$\begin{bmatrix} \beta_1^{(0,0)} \\ \beta_1^{(1,0)} \\ \beta_1^{(0,1)} \\ \beta_1^{(1,1)} \end{bmatrix} = \left(\Gamma_{\frac{1}{2},0}\right)^{-1} \begin{bmatrix} \frac{1}{2}\varphi(\theta)_{\frac{1}{2},0}^h \\ \frac{1}{2}\varphi(\theta)_{-\frac{1}{2},0}^h \\ \frac{1}{2}\tilde{c}_2\varphi(\theta)_{\frac{1}{2},0}^h \\ \frac{1}{2}\tilde{c}_2\varphi(\theta)_{-\frac{1}{2},0}^h \end{bmatrix} = \frac{1}{4} \begin{bmatrix} c_1(1+\tilde{c}_2) \\ s_1(1+\tilde{c}_2) \\ c_1(1-\tilde{c}_2) \\ s_1(1-\tilde{c}_2) \end{bmatrix} \quad (7.17)$$

Note that the trigonometric functions figuring in (7.17) are evaluated for  $\theta = \theta^{(0,0)}$ . Using the trigonometric identities  $1 + \cos(2\alpha) = 2 \cos^2(\alpha)$ ,  $\cos(\alpha + \frac{\pi}{2}) = -\sin(\alpha)$  and  $\cos(\alpha + \pi) = -\cos(\alpha)$ , (7.17) can be recast into a general formula:

$$\beta_1^{(a,b)} = \frac{1}{2}(-1)^a c_{1,a} c_{2,b}^2, \quad a, b \in \{0, 1\}. \quad (7.18)$$

Similarly, we have for the vertical prolongation:

$$\beta_2^{(a,b)} = \frac{1}{2}(-1)^b c_{2,b} c_{1,a}^2, \quad a, b \in \{0, 1\}. \quad (7.19)$$

The formulae (7.18) and (7.19) can be combined into the format of (3.3):

$$P\tilde{\Phi}(2\theta^{(0,0)}) \begin{bmatrix} \tilde{\alpha}_1 \\ \tilde{\alpha}_2 \end{bmatrix} = \sum_{a,b \in \{0,1\}} \Phi(\theta^{(a,b)}) \begin{bmatrix} \alpha_1^{(a,b)} \\ \alpha_2^{(a,b)} \end{bmatrix} \quad (7.20)$$

with

$$\begin{bmatrix} \alpha_1^{(a,b)} \\ \alpha_2^{(a,b)} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} (-1)^a c_{1,a} c_{2,b}^2 & 0 \\ 0 & (-1)^b c_{2,b} c_{1,a}^2 \end{bmatrix} \begin{bmatrix} \tilde{\alpha}_1 \\ \tilde{\alpha}_2 \end{bmatrix} := P(\theta^{(a,b)}) \begin{bmatrix} \tilde{\alpha}_1 \\ \tilde{\alpha}_2 \end{bmatrix}$$

Using (7.12), (7.20) and (7.21) can be transformed into:

$$P\Phi_H(2\theta^{(0,0)}) \begin{bmatrix} \tilde{\alpha}_g \\ \tilde{\alpha}_s \end{bmatrix} = \sum_{a,b \in \{0,1\}} \Phi_H(\theta^{(a,b)}) \begin{bmatrix} \alpha_g^{(a,b)} \\ \alpha_s^{(a,b)} \end{bmatrix}, \quad (7.21)$$

with

$$\begin{aligned} \begin{bmatrix} \alpha_g^{(a,b)} \\ \alpha_s^{(a,b)} \end{bmatrix} &= \left( H(\theta^{(a,b)}) \right)^{-1} P(\theta^{(a,b)}) H(2\theta^{(0,0)}) \begin{bmatrix} \tilde{\alpha}_g \\ \tilde{\alpha}_s \end{bmatrix} \\ &= \begin{bmatrix} c_{1,a}^2 c_{2,b}^2 & (-1)^{a+b} \tilde{s}_{1,a} \tilde{s}_{2,b} \frac{c_{2,b}^2 - c_{1,a}^2}{4(\tilde{s}_{1,a}^2 + \tilde{s}_{2,b}^2)} \\ 0 & (-1)^{a+b} c_{1,a} c_{2,b} \frac{\tilde{s}_{1,a}^2 + \tilde{s}_{2,b}^2}{4(\tilde{s}_{1,a}^2 + \tilde{s}_{2,b}^2)} \end{bmatrix} \begin{bmatrix} \tilde{\alpha}_g \\ \tilde{\alpha}_s \end{bmatrix} := P_H(\theta^{(a,b)}) \begin{bmatrix} \tilde{\alpha}_g \\ \tilde{\alpha}_s \end{bmatrix} \end{aligned} \quad (7.22)$$

*Remark.* The nonzero structure of  $P_H(\theta^{(a,b)})$  shows that the prolongation of a coarse gradient vector is a fine gradient vector. This corresponds to the node-edge commutation property (1.4).

*Remark.* Similarly as for standard nodal prolongation and restriction, the 4-frequency representation of the restriction operator (7.10) is equal the scaled transpose of the 4-frequency representation of the prolongation operator (7.21):

$$\begin{bmatrix} R(\theta^{(0,0)}) & R(\theta^{(1,0)}) & R(\theta^{(0,1)}) & R(\theta^{(1,1)}) \end{bmatrix} = 4 \begin{bmatrix} P(\theta^{(0,0)}) \\ P(\theta^{(1,0)}) \\ P(\theta^{(0,1)}) \\ P(\theta^{(1,1)}) \end{bmatrix}^T.$$

Taking into account that  $\tilde{s}_{i,a}^2 = \tilde{s}_i^2$  for  $a \in \{0, 1\}$ , the formulae (7.22) and (7.14) are related as:

$$R_H(\theta^{(a,b)}) = 4 \frac{\tilde{s}_{1,a}^2 + \tilde{s}_{2,b}^2}{\tilde{s}_1^2 + \tilde{s}_2^2} \left( P_H(\theta^{(a,b)}) \right)^T = 4 \frac{\det(S(\theta^{(a,b)}))}{\det(\tilde{S}(2\theta^{(0,0)}))} \left( P_H(\theta^{(a,b)}) \right)^T.$$

The additional scaling factor arises from the fact that the transformation matrices  $H(\theta^{(a,b)})$  and  $H(2\theta^{(0,0)})$  are not normalized.

**8. Multigrid cycle.** We consider a standard 2-level multigrid cycle, which consists of the following steps (cfr. [13]):

1.  $\mu_{pre}$  presmoothing steps on  $Kx = b$
2. restriction to coarse space:  $b_c = R(b - Kx)$
3. coarse solve:  $K_c x_c = b_c$
4. prolongation to fine space:  $x \leftarrow x + P x_c$
5.  $\mu_{post}$  postsmoothing steps on  $Kx = b$

with  $R$ ,  $P$  and  $K_c$  denoting the restriction matrix, the prolongation matrix and the coarse system matrix. We suppose the coarse system matrix  $K_c$  to be built as the Galerkin product  $RKP$ .

Due to the aliasing of the harmonic frequencies  $\theta^{(a,b)}$  of  $\theta \in \Theta^{\text{low}}$  on the coarse grid (see section 7.1), the multigrid cycle needs to be studied in  $F_8(\theta)$ . For this purpose, the Fourier domain representation of the multigrid cycle can be constructed from its constituent parts, similarly as in [13]:

$$\mathcal{C}(\theta) = \mathcal{S}_{post}(\theta)^{\mu_{post}} \left( I_8 - \mathcal{P}(\theta) \mathcal{K}_c(\theta)^{-1} \mathcal{R}(\theta) \mathcal{K}(\theta) \right) \mathcal{S}_{pre}(\theta)^{\mu_{pre}}. \quad (8.1)$$

Here,  $\mathcal{K}(\theta)$ ,  $\mathcal{S}_{pre}(\theta)$ ,  $\mathcal{S}_{post}(\theta) \in \mathbb{C}^{8 \times 8}$  represent the operator  $K$  and the pre- and post-smoothing operators in  $F_8(\theta)$  with respect to the parameterization (3.15). The re-

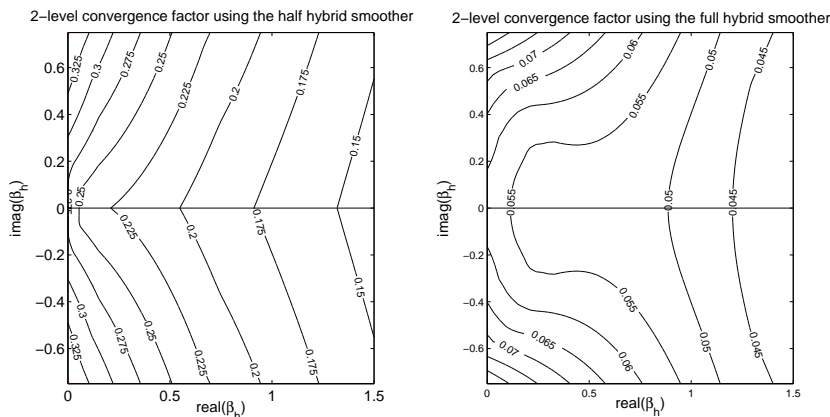


FIG. 8.1. Asymptotic convergence factors as a function of  $\beta$  for a 2-level multigrid cycle with  $\mu_{pre} = \mu_{post} = 1$  using the half hybrid smoother (left) and the full hybrid smoother (right) consisting of node-based lexicographic Gauss-Seidel iterations in edge space and lexicographic Gauss-Seidel iterations in nodal space.

striction and prolongation operators and the coarse system matrix are defined as:

$$\begin{aligned} \mathcal{R}(\theta) &:= \begin{bmatrix} R(\theta^{(0,0)}) & R(\theta^{(1,0)}) & R(\theta^{(0,1)}) & R(\theta^{(1,1)}) \end{bmatrix} \in \mathbb{C}^{2 \times 8} \\ \mathcal{P}(\theta) &:= \begin{bmatrix} P(\theta^{(0,0)}) & P(\theta^{(1,0)}) & P(\theta^{(0,1)}) & P(\theta^{(1,1)}) \end{bmatrix}^T \in \mathbb{C}^{8 \times 2} \\ \mathcal{K}_c(\theta) &:= \mathcal{R}(\theta)\mathcal{K}(\theta)\mathcal{P}(\theta) \in \mathbb{C}^{2 \times 2}. \end{aligned}$$

Now, the asymptotic convergence factor can be calculated as

$$\rho_\infty = \sup\{\rho(\mathcal{C}(\theta)) : \theta \in \Theta^{\text{low}}\}. \quad (8.2)$$

Figures 8.1 and 8.2 show the asymptotic convergence factors of a 2-level multigrid cycle using the hybrid smoother and the AFW-smoother. Note that the LFA allows to analyze multigrid for the curl-curl equation for *any* value of  $\beta$ . Of particular interest are the purely real and purely imaginary cases, depicted in Figure 8.3. The latter corresponds to the time-harmonic curl-curl equation. The former arises for example in the backward Euler discretization of the time-dependent curl-curl equation.

*Remark.* For  $\theta = (0, 0)$  and  $\beta = 0$ , the above derivation does not hold, because  $\mathcal{K}_c(\theta)$  is singular. This is a consequence of the absence of boundary conditions, which is generic for a *local* Fourier analysis (cfr. [13], p.109). To avoid this complication,  $\beta = 0$  has been replaced by  $\beta = 10^{-6}$  to generate the Figures 8.1 and 8.2.

**9. Remarks about the numerical tests.** All numerical and analytical results and all figures in this paper have been obtained for the discretization of the two-dimensional curl-curl equation (1.1) on a square with periodic boundary conditions. A local Fourier analysis yields exact results for this problem setting (see [13]). In the numerical tests, the Gauss-Seidel smoothers of section 6.1 are not implemented as a sequence of local solves for each edge  $e$

$$x(e) = \frac{b(e) - \sum_{c \neq e} K(e, c)x(c)}{K(e, e)},$$

because this implies the use of *different* update stencils at the boundaries. This does not correspond to the smoother as analyzed in the LFA. This inconsistency is avoided

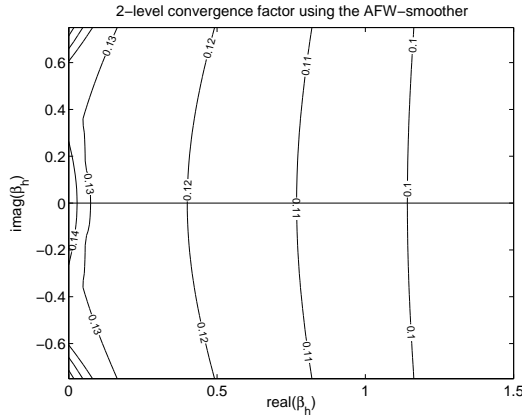


FIG. 8.2. Asymptotic convergence factors as a function of  $\beta$  for a 2-level multigrid cycle with  $\mu_{pre} = \mu_{post} = 1$  using the AFW-smoother.

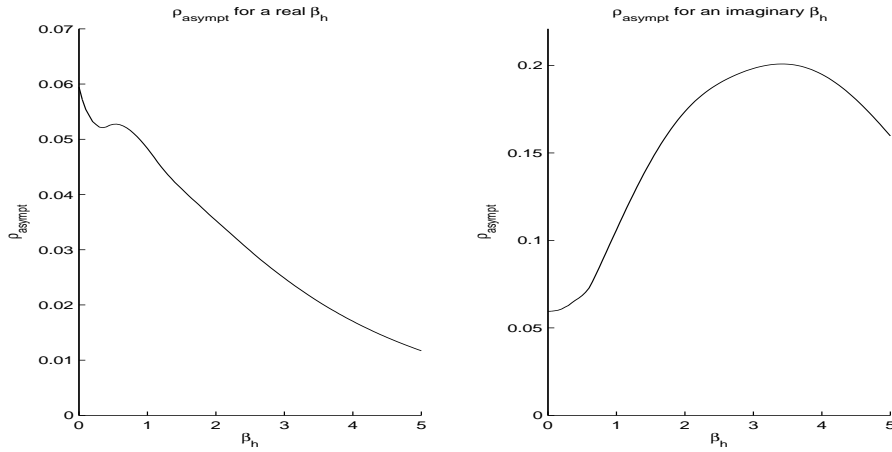


FIG. 8.3. Asymptotic convergence factors as a function of a purely real (left) and a purely imaginary (right)  $\beta_h$  for a 2-level multigrid cycle with  $\mu_{pre} = \mu_{post} = 1$  using the full hybrid smoother consisting of node-based lexicographic Gauss-Seidel iterations in edge space and lexicographic Gauss-Seidel iterations in nodal space.

by implementing the Gauss-Seidel smoothers as  $x \leftarrow x + \tilde{K}^{-1}(b - Kx)$ , with the nonzero pattern of  $\tilde{K}$  based on the update stencils of Figure 6.1:

$$\tilde{K}(r, c) = 0, \quad \text{except} \quad \tilde{K}(e, [e \text{ new}(e)]) = K(e, [e \text{ new}(e)]).$$

Here,  $\text{new}(e)$  are the indices of the edges whose *updated* values are to be used in the update formula for the edge  $e$  instead of their old values. Note that the resulting smoothing procedure is *implicit* (cfr. Remark 4.6.4. in [13]), since it is not possible any more to transform this  $\tilde{K}$  into a triangular matrix by means of a permutation. All asymptotic convergence factors calculated by means of the LFA in this paper were checked numerically by applying 2-level multigrid iterations for a zero righthand side and a random initial guess. This procedure benefits from the non-normalized floating point representation, which allows to reduce the norm of the residual much more than

TABLE 9.1

Convergence history of 2-level  $V(1,1)$  multigrid iterations using the full hybrid smoother for the finite element discretization of the two-dimensional curl-curl equation (1.1) with  $\beta = 0.1$  on a quadrilateral grid with  $24 \times 24$  nodes. The boundary conditions are periodic. The righthand side is zero and the initial guess is random. The analytical convergence factor  $\rho_{anal}$  is calculated using LFA, the numerical convergence factor is calculated as  $\rho_{num}^{[\nu]} = \|b - Kx^{[\nu]}\| / \|b - Kx^{[\nu-1]}\|$ .

iteration	$\ b - Kx\ $	$\rho_{num}$	$\frac{ \rho_{num} - \rho_{anal} }{\rho_{anal}}$
190	$9.21065e - 240$	0.055395131977695	0.003494%
191	$5.10226e - 241$	0.055395193735180	0.003382%
192	$2.82641e - 242$	0.055395253466549	0.003274%
193	$1.56570e - 243$	0.055395311241170	0.003170%
194	$8.67324e - 245$	0.055395367125882	0.003069%
195	$4.80458e - 246$	0.055395421185096	0.002971%
196	$2.66152e - 247$	0.055395473480890	0.002877%
197	$1.47436e - 248$	0.055395524073103	0.002786%
198	$8.16731e - 250$	0.055395573019418	0.002697%
199	$4.52433e - 251$	0.055395620375452	0.002612%
200	$2.50628e - 252$	0.055395666194833	0.002529%

with a random righthand side. For all results presented in this paper, it was possible to reduce the relative difference between the analytical and numerical values for the convergence factors to 0.1% by increasing the number of iterations. Table 9.1 shows an excerpt of the convergence history of a 2-level  $V(1,1)$  multigrid iteration using the full hybrid smoother.

**10. Generalization.** The separate treatment of edges with different orientations, as introduced in section 3, can be applied to any translation-invariant set of stencils on a translation-invariant grid. More precisely, it can be applied when it is possible to represent the grid by means of a *single* node-edge pattern (see Figure 10.1) and the operator by a stencil depending *only* on the orientation of the edge. In this case, Theorems 3.1 and 3.2 can be generalized.

For instance, the regular two-dimensional triangular mesh of Figure 10.1 contains 3 different types of edges. Correspondingly, 3 stencils will be needed to represent a discrete edge operator  $L^e$  and each frequency will be related to a three-dimensional eigenspace of  $L^e$ . Similarly as (3.6), the decoupled basis for this space corresponds to the columns of:

$$\Phi(\theta) = \begin{bmatrix} \varphi(\theta, \mathbf{x})_{E_1} & 0_{E_1} & 0_{E_1} \\ 0_{E_2} & \varphi(\theta, \mathbf{x})_{E_2} & 0_{E_2} \\ 0_{E_3} & 0_{E_3} & \varphi(\theta, \mathbf{x})_{E_3} \end{bmatrix}.$$

In a similar way, this approach can be generalized for face-based problems in  $H(\text{div})$  represented by translation-invariant stencils on a translation-invariant grid.

**11. Conclusion.** We presented a local Fourier analysis for multigrid for the finite element discretization of the two-dimensional curl-curl equation on a quadrilateral grid. Our approach is based on the decoupling in the Fourier domain of edges with different orientation. In this way, each frequency is related to a two-dimensional space. This procedure allows to study the smoothing properties of multigrid smoothers and the asymptotic convergence behavior of the multigrid cycle. We identified in the Fourier domain several properties which are important in the context of the curl-curl

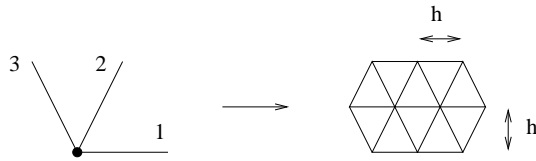


FIG. 10.1. two-dimensional triangular mesh generated by a single node-edge pattern for each node. The node-edge pattern relates each node to edges of 3 different orientations.

equation, such as the Helmholtz splitting, topological identities and the node-edge and edge-face commutation properties. We showed how our approach can be extended to some other types of translation-invariant meshes and to translation-invariant problems in  $H(\text{curl})$  and  $H(\text{div})$ .

#### REFERENCES

- [1] D. Arnold, R. Falk, and R. Winther. Multigrid in  $H(\text{div})$  and  $H(\text{curl})$ . *Numerische Mathematik*, 85:197–218, 2000.
- [2] P. Bochev, C. Garasi, J. Hu, A. Robinson, and R. Tuminaro. An improved algebraic multigrid method for solving Maxwell’s equations. *SIAM Journal on Scientific Computing*, 25:623–642, 2003.
- [3] T. Boonen, G. Deliége, and Vandewalle. On algebraic multigrid methods derived from partition of unity nodal prolongators. *Numerical Linear Algebra with Applications*, 13(2-3)(2-3):105–131, 2006.
- [4] A. Bossavit. *Computational Electromagnetism*. Academic Press, Boston, 1998.
- [5] A. Brandt. Multi-level adaptive solutions to boundary-value problems. *Math. Comp.*, 31:333–390, 1977.
- [6] E. Chung and J. Zou. The eigenvalues and eigenspaces of some discrete div- and curl-related operators. *SIAM Journal on Matrix Analysis and Applications*, 24(4):1149–1160, 2003.
- [7] R. Hiptmair. Multigrid method for Maxwell’s equations. *SIAM Journal on Numerical Analysis*, 36(1):204–255, 1999.
- [8] J. Hu, R. Tuminaro, P. Bochev, C. Garasi, and A. Robinson. Toward an h-independent algebraic multigrid method for Maxwell’s equations. *SIAM Journal on Scientific Computing*, 27(5):1669–1688, 2006.
- [9] P. Monk. *Finite element methods for Maxwell’s equations*. Clarendon Press, Oxford, 2003.
- [10] R. Perrussel. *Méthodes multiniveaux algébriques pour les éléments d’arête. Applications à l’électromagnétisme*. PhD thesis, École Centrale de Lyon, Lyon, France, October 2005.
- [11] SANDIA National Laboratory (US). *ML 3.1 Smoothed Aggregation User’s Guide*.
- [12] K. Stueben. A review of algebraic multigrid. *Journal of Computational and Applied Mathematics*, 128:281–309, 2001.
- [13] U. Trottenberg, C. Oosterlee, and A. Schuller. *Multigrid*. Academic Press, London, 2001.
- [14] P. Wesseling. *An introduction to Multigrid Methods*. John Wiley, Chichester, 1992.
- [15] Roman Wienands and Wolfgang Joppich. *Practical Fourier analysis for multigrid methods*. Chapman and Hall/CRC Press, 2005.