Applications of Mathematics

Radim Blaheta; P. Byczanski; Roman Kohut

Composite grid finite element method: Implementation and iterative solution with inexact subproblems

Applications of Mathematics, Vol. 47 (2002), No. 2, 83–100

Persistent URL: http://dml.cz/dmlcz/134487

47 (2002) APPLICATIONS OF MATHEMATICS No. 2, 83–100

COMPOSITE GRID FINITE ELEMENT METHOD: IMPLEMENTATION AND ITERATIVE SOLUTION WITH INEXACT SUBPROBLEMS*

R. Blaheta, P. Byczanski and R. Kohut, Ostrava

Abstract. This paper concerns the composite grid finite element (FE) method for solving boundary value problems in the cases which require local grid refinement for enhancing the approximating properties of the corresponding FE space. A special interest is given to iterative methods based on natural decomposition of the space of unknowns and to the implementation of both the composite grid FEM and the iterative procedures for its solution. The implementation is important for gaining all benefits of the described methods. We also discuss the case of inexact subproblems, which can frequently arise in the course of hierarchical modelling.

Keywords: finite element method, composite grids, iterative solution, computer implementation, inexact subproblems, numerical experiments

MSC 2000: 65N30, 65N22, 65F10

1. Introduction

This paper concerns the numerical solution of elliptic boundary value problems, which can be written in the weak form

$$\text{find } u \in \mathcal{V}\colon\ a(u,v) = b(v)\ \ \forall v \in \mathcal{V} \tag{1.1}$$

where $\mathcal{V}$ is a Hilbert space of functions defined on a domain Ω, a is a bounded symmetric positive definite bilinear form on $\mathcal{V}$ and b is a bounded linear functional on $\mathcal{V}$. See e.g. [1], [6] for more details.

* This work has been supported by the Grant INCO KIT 977006 of the European Commission and the Grant OK 383 of the Ministry of Education of the Czech Republic.

For an accurate and efficient numerical solution of the problem (1.1) by the finite element (FE) method, it is important

- to use FE grids which are locally refined in some part $\Omega_R \subset \Omega$,
- to facilitate an adaptive construction of Ω_R, which need not be known a priori,
- to find an efficient solver for the FE systems,
- to make it possible to use efficient data structures as those arising in the case of regular grids,
- to decompose the numerical solution into subtasks which are suitable for faster data transfers and parallelization of the computations.

The basic requirement for the local grid refinement can be satisfied by the use of unstructured locally refined FE grids or by the use of composite FE grids, which arise as a composition of a (regular, structured) global grid and (regular) local grid(s).

The *composite grid FE method* starts with a global (coarse, regular) division $\mathcal{T}_H(\Omega)$ of the domain Ω. For simplicity, we will assume a division of the domain $\Omega \subset R^d$ into triangles (for $d = 2$) or tetrahedrons (for $d = 3$). The division $\mathcal{T}_H(\Omega)$ then makes it possible to define a standard FE space

$$V_H(\Omega) = \{v \in C(\bar{\Omega}) \cap \mathcal{V}\colon\ v|_E \in P_1\ \ \forall E \in \mathcal{T}_H(\Omega)\} \tag{1.2}$$

where P_1 is in our case the set of linear polynomials.

Further, let $\Omega_R \subset \Omega$ be a part of Ω where the above discretization should be refined and let $\mathcal{T}_h = \mathcal{T}_h(\Omega_R)$ be a finer discretization in Ω_R, which enables us to define another FE space

$$V_h(\Omega) = \{v \in C(\bar{\Omega}) \cap \mathcal{V}\colon\ v \equiv 0 \text{ in } \Omega\backslash\Omega_R,\ \ v|_E \in P_1\ \ \forall E \in \mathcal{T}_h(\Omega_R)\}. \tag{1.3}$$

Now, the composite grid FE space can be defined as

$$V = V_0 + V_1, \quad V_0 = V_H(\Omega), \quad V_1 = V_h(\Omega). \tag{1.4}$$

The composite grid FE space $V \subset \mathcal{V}$ can be then used for finding the approximate composite grid solution

$$u \in V\colon\ a(u, v) = b(v)\ \ \forall v \in V. \tag{1.5}$$

Note that, in this construction, Ω_R can consist of several separate parts. Moreover, the finer grid in Ω_R can be further refined and so on, which gives a multi-level construction resulting in the composite grid space $V = V_0 + V_1 + \ldots + V_p$.

Further, we would like to show the potential of the composite grid FE method to satisfy the described attributes of the efficient finite element method.

The organization of this paper is the following. In Section 2, we describe iterative methods which naturally arise from the existing decomposition (1.4) of the composite grid FE space. These methods facilitate also the decomposition and parallelization of the solution of the composite grid problems. Section 3 is devoted to the implementation of both the composite grid FE method and the iterative solution methods. Section 4 is devoted to the case when the coarse grid problem is not fully compatible with the composite grid problem in a part of Ω_R. In the last section, we present a numerical example for illustration of the behaviour of the described methods.

2. Iterative solution of composite grid problems

The decomposition (1.4) of the composite grid FE space can be used for the construction of iterative methods for solving the composite grid problem (1.5).

The first method of this type was first introduced in [13], see also [14]. The algorithm of this method, called FAC, is the following:

FAC method
given u^0
for $i = 0, 1 \ldots$ **until** convergence **do**
compute $v_0 \in V_0$: $a(u^i + v_0, v) = b(v) \ \forall v \in V_0$
$u^{i+1/2} = u^i + v_0$
compute $v_1 \in V_1$: $a(u^{i+1/2} + v_1, v) = b(v) \ \forall v \in V_1$
$u^{i+1} = u^{i+1/2} + v_1$
end

Further, we will use also a symmetric variant of FAC, which arises by adding another correction from the second subspace to the beginning of each iteration. This variant will be called SFAC.

An additive variant of FAC was introduced in [10], [15], see also [14], as the AFAC method. For ensuring the convergence in the case that $V_0 \cap V_1 \neq \{0\}$, the multiple contribution from the overlap of the subspaces V_0 and V_1 is corrected. The algorithm is the following:

AFAC method
given u^0
for $i = 0, 1 \ldots$ **until** convergence **do**
compute $v_0 \in V_0$: $a(u^i + v_0, v) = b(v) \ \forall v \in V_0$
compute $v_1 \in V_1$: $a(u^i + v_1, v) = b(v) \ \forall v \in V_1$
compute $w \in V_0 \cap V_1$: $a(u^i + w, v) = b(v) \ \forall v \in V_0 \cap V_1$
$u^{i+1} = u^i + v_0 + v_1 - w$
end

An advantage of AFAC is a higher possibility of parallelization in the algorithm. Another additive variant of FAC called JFAC (as a Jacobi-type method with damping), can be introduced in the following form:

JFAC method
given u^0
for $i = 0, 1 \ldots$ **until** convergence **do**
compute $v_0 \in V_0$: $a(u^i + v_0, v) = b(v) \ \forall v \in V_0$
compute $v_1 \in V_1$: $a(u^i + v_1, v) = b(v) \ \forall v \in V_1$
$u^{i+1} = u^i + \frac{1}{2} v_0 + \frac{1}{2} v_1$
end

The convergence of JFAC can be expected to be somewhat worse as compared with AFAC, but the implementation of JFAC is simpler and the iteration of JFAC is cheaper.

All these methods can be rewritten into the operator form which is closer to the computer implementation of the methods. For this purpose, we shall use an inner product $\langle , \rangle$ in V and define for $i = 0, 1, 01$:

$$A\colon V \to V \quad \text{by} \quad \langle Au, v\rangle = a(u, v) \quad \forall u, v \in V, \tag{2.1}$$

$$b \in V \quad \text{by} \quad \langle b, v\rangle = b(v) \quad \forall v \in V, \tag{2.2}$$

$$A_i\colon V_i \to V_i \quad \text{by} \quad \langle A_i u, v\rangle = a(u, v) \quad \forall u, v \in V_i, \tag{2.3}$$

$$b_i \in V_i \quad \text{by} \quad \langle b_i, v\rangle = b(v) \quad \forall v \in V_i. \tag{2.4}$$

Further, $V_{01} = V_0 \cap V_1$. For $i = 0, 1, 01$, we define also the inclusion and restriction operators,

$$I_i\colon V_i \to V, \quad R_i\colon V \to V_i. \tag{2.5}$$

As the restriction, we shall take the $\langle , \rangle$-orthogonal projection from V to V_i. Then it is easy to prove the following lemma.

Lemma 2.1. *For $i = 0, 1, 01$, R_i is the adjoint operator to I_i and*

$$A_i = R_i\, A\, I_i, \quad b_i = R_i\, b. \tag{2.6}$$

Moreover, for $v \in V$, $v \neq 0$ at least one of $R_0 v$, $R_1 v$ is nonzero.

P r o o f. As R_i is the $\langle , \rangle$-orthogonal projection $V \to V_i$, we have

$$\langle R_i u, v\rangle = \langle R_i u + u - R_i u, v\rangle = \langle u, v\rangle = \langle u, I_i v\rangle$$

for any $u \in V$, $v \in V_i$. This shows $R_i = I_i^*$.

Now, it is simple to prove (2.6), because for any $u, v \in V_i$ we get

$$b(v) = \langle b, v\rangle = \langle b, I_i v\rangle = \langle R_i b, v\rangle,$$
$$a(u, v) = \langle Au, v\rangle = \langle AI_i u, I_i v\rangle = \langle R_i A I_i u, v\rangle.$$

Finally, let $v \in V$, $v \neq 0$ and let both $R_0 v, R_1 v$ be zero. Then for any $w \in V$, $w = w_0 + w_1$, $w_i \in V_i$, we get

$$\langle v, w\rangle = \langle v, I_0 w_0 + I_1 w_1\rangle = \langle R_0 v, w_0\rangle + \langle R_1 v, w_1\rangle = 0.$$

This contradicts the assumption $v \neq 0$. Hence, one of $R_0 v, R_1 v$ must be nonzero. □

The composite grid FE problem can be now rewritten as the equation

$$Au = b \tag{2.7}$$

with a symmetric positive definite operator A. The iterative methods FAC, AFAC and JFAC can be written as preconditioned Richardson's iterations

$$u^{i+1} = u^i + G(b - Au^i) \tag{2.8}$$

with

$$G = G_{FAC} = B_0 + B_1 - B_1 A B_0 \quad \text{for the FAC method,} \tag{2.9}$$
$$G = G_{AFAC} = B_0 + B_1 - B_{01} \quad \text{for the AFAC method,} \tag{2.10}$$
$$G = G_{JFAC} = \frac{1}{2}B_0 + \frac{1}{2}B_1 \quad \text{for the JFAC method.} \tag{2.11}$$

Here

$$B_i = I_i\, A_i^{-1}\, R_i. \tag{2.12}$$

As A is symmetric positive definite, the system (2.7) can be also solved by the conjugate gradient method with the above preconditioners. Nonsymmetric G_{FAC} can be replaced by the symmetrized FAC preconditioner introduced for the first time in [7]. This preconditioner has the form

$$G_{\mathrm{SFAC}} = B_1 + B_0 - B_1 A B_0 - B_0 A B_1 + B_1 A B_0 A B_1. \tag{2.13}$$

It can be implemented as the approximate solution of the system $Ag = r$ by one SFAC iteration (one-and-half FAC iterations) starting from the zero initial guess.

Note that the iteration matrices of FAC, AFAC and JFAC can be written in the form

$$M_{\mathrm{FAC}} = (I - P_1)(I - P_0), \tag{2.14}$$

$$M_{\mathrm{SFAC}} = (I - P_1)(I - P_0)(I - P_1), \tag{2.15}$$

$$M_{\mathrm{AFAC}} = I - (P_0 + P_1 - P_{01}), \tag{2.16}$$

$$M_{\mathrm{JFAC}} = I - \frac{1}{2}(P_0 + P_1), \tag{2.17}$$

where $P_i = B_i A = I_i A_i^{-1} R_i A$ are projections $V \to V_i$, $i = 0, 1, 01$, which are orthogonal with respect to the inner product

$$\langle u, v \rangle_A = \langle Au, v \rangle \ \ \forall u, v \in V. \tag{2.18}$$

3. Implementation

In this section, we focus our attention on the implementation of the two-level composite grid FE method arising from the spaces V_H and V_h. We shall start the implementation from introducing the standard FE bases $\{\Phi_i^H\}$, $\{\Phi_i^h\}$ in V_H and V_h, respectively. Then we can introduce the subproblem matrices and right-hand side vectors as

$$A_0 = A_H = \left[a(\Phi_i^H, \Phi_j^H)\right], \quad b_0 = b_H = \left[b(\Phi_j^H)\right], \quad i, j = 1, \ldots, n_H, \tag{3.1}$$

$$A_1 = A_h = \left[a(\Phi_i^h, \Phi_j^h)\right], \quad b_1 = b_h = \left[b(\Phi_i^h)\right], \quad i, j = 1, \ldots, n_h. \tag{3.2}$$

Original bases (OB) implementation. For implementation of the composite grid FE method, we must introduce some parametrization of the functions from the composite grid FE space V. The first possibility is to use a straightforward representation

$$u \in V \Rightarrow u = \sum_i u_{H,i} \Phi_i^H + \sum_i u_{h,i} \Phi_i^h. \tag{3.3}$$

The coefficients $\bar{u} = (\ldots u_{H,i} \ldots, \ldots u_{h,i} \ldots)^T = (u_H, u_h)^T$ of the composite grid FE solution will solve the Ritz-Galerkin algebraic system $Au = b$,

$$Au = b \Leftrightarrow \begin{bmatrix} A_{HH} & A_{Hh} \\ A_{hH} & A_{hh} \end{bmatrix} \begin{bmatrix} u_H \\ u_h \end{bmatrix} = \begin{bmatrix} b_H \\ b_h \end{bmatrix}, \tag{3.4}$$

where $A_{HH} = A_H$, $A_{hh} = A_h$, b_H and b_h are defined in (3.1) and (3.2), respectively. The only additional work to the assembling of the standard FE systems (3.1) and (3.2) is the computation of $A_{Hh} = [a(\Phi_i^H, \Phi_j^h)]$. The remaining block A_{hH} is equal to the transpose A_{Hh}^T.

Up to now, the composite grid FE method has not used any relation between the elements of $\mathcal{T}_H$ and $\mathcal{T}_h$. But some relation between them can be exploited for the computation of A_{Hh}.

We say that the divisions $\mathcal{T}_H$ and $\mathcal{T}_h$ are *fully compatible* if the elements of $\mathcal{T}_h$ arise as divisions of the elements of $\mathcal{T}_H$. In this case, for any basis function Φ_i^H,

$$\Phi_i^H|_{\Omega_R} = \sum_k \varphi_{ik} \Psi_k^h, \tag{3.5}$$

where Ψ_k^h are the standard basis functions corresponding to $\mathcal{T}_h(\Omega_R)$. Note that Ψ_k^h need not vanish on the inner boundary $\partial\Omega_R \setminus \partial\Omega$, thus we have the relations $\{\Phi_i^h\} = \{\Psi_k^h\colon\ \Psi_k^h = 0 \text{ on } \partial\Omega_R \setminus \partial\Omega\} \subset \{\Psi_k^h\}$.

If (3.5) holds, then A_{hh}, A_{Hh} and A_{hH} can be constructed from

$$\overline{A}_{hh} = \left[a(\Psi_i^h, \Psi_j^h)\right].$$

Let $I_h^H = [\varphi_{ik}]$ and let I_B be the operator which extends the vectors $v_h \in V_h$ by zeros corresponding to the fine grid nodes on $\partial\Omega_R \setminus \partial\Omega$. Note that we assume that Dirichlet's boundary conditions on a part of $\partial\Omega$ are already treated, if not then we would take the zero extension also to this part of the boundary. Then

$$A_{hh} = I_B^T \overline{A}_{hh} I_B, \quad A_{Hh} = I_h^H \overline{A}_{hh} I_B. \tag{3.6}$$

Note that it is also possible to construct A_{HH} as

$$A_{HH} = A_{HH}^- + I_h^H \overline{A}_{hh} (I_h^H)^T$$

where A_{HH}^- is the matrix assembled from the contributions of those elements of $\mathcal{T}_H$ which lie outside Ω_R.

In the described case of fully compatible global and local grids, the expression (3.3) is not unique and the system (3.4) is therefore singular, but consistent. This singularity does not mind when solving (3.4) by the iterative methods described in Section 2.

Nodal basis (NB) implementation. In the case of fully compatible global and local grids, it is also possible to use another implementation based on a nodal

FE basis of the composite grid FE space. It means that $u \in V$ is represented in the form

$$u = \sum u_{H,i}\Phi_i^H + \sum u_{h,i}\Phi_i^h + \sum u_{\mathcal{H},i}\Phi_i^{\mathcal{H}} \tag{3.7}$$

where the first sum is over that coarse grid basis functions Φ_i^H which have their support in $\Omega \setminus \Omega_R$. The second sum is over all basis functions from V_h. The third sum is over new basis functions corresponding to the coarse grid nodes lying on the interface $\partial\Omega_R \setminus \partial\Omega$. The basis functions $\Phi_i^{\mathcal{H}} \subset V$ have zero values in all coarse grid and fine grid nodes with the exception of value 1 in one coarse grid node on $\partial\Omega_R \setminus \partial\Omega$ and the values given by interpolation in the corresponding fine grid nodes on $\partial\Omega_R \setminus \partial\Omega$. This interpolation guarantees continuity of the basis function $\Phi_i^{\mathcal{H}}$. See also Fig. 1.

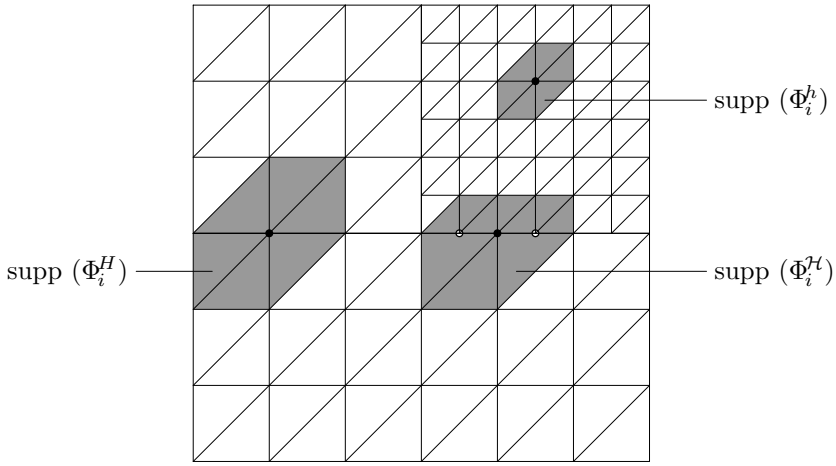

Figure 1. Nodal basis of the composite grid FE space.

The FE system $Au = b$ corresponding to NB implementation is regular symmetric positive definite with the block structure

$$\begin{bmatrix} \tilde{A}_{HH} & 0 & A_{H\mathcal{H}} \\ 0 & A_{hh} & A_{h\mathcal{H}} \\ A_{\mathcal{H}H} & A_{\mathcal{H}h} & A_{\mathcal{H}\mathcal{H}} \end{bmatrix} \begin{bmatrix} \tilde{u}_H \\ u_h \\ u_{\mathcal{H}} \end{bmatrix} = \begin{bmatrix} \tilde{b}_H \\ b_h \\ b_{\mathcal{H}} \end{bmatrix}, \tag{3.8}$$

where A_{hh} is the full block A_h from (3.2), $\tilde{A}_{HH}$ is a submatrix of A_H from (3.1). The product of A with a vector $v = (\tilde{v}_H, v_h, v_{\mathcal{H}})^T$ can be computed from $A_H v_H$, where $v_H \in V_H(\Omega)$ is a zero extension of $\tilde{v}_H, A_{hh}v_h$ and the remaining part, which can be assembled from the element contributions, see [16].

Note that this NB implementation in principle does not require compatibility of the global and local grids in Ω_R, which is expressed by (3.5). But in the case of not fully compatible global and local grids, we seek the composite grid solution in the space $\tilde{V} \neq V$, where $\tilde{V}$ is the space of the vectors that have the form (3.7).

Hierarchical basis (HB) implementation. In the case of fully consistent global and local grids, there is still another way of implementation based on a hierarchical basis, which consists of the coarse grid basis functions $\{\Phi_i^H\}$ and those basis functions from $\{\Phi_i^h\}$ which correspond to the fine grid nodes which are not the coarse grid nodes.

The corresponding FE system is then again regular with the block structure

$$\begin{bmatrix} A_{HH} & \tilde{A}_{Hh} \\ \tilde{A}_{hH} & \tilde{A}_{hh} \end{bmatrix} \begin{bmatrix} u_H \\ \tilde{u}_h \end{bmatrix} = \begin{bmatrix} b_H \\ \tilde{b}_h \end{bmatrix} \tag{3.9}$$

where A_{HH} is the full block A_H from (3.1) and $\tilde{A}_{hh}$, $\tilde{A}_{Hh}$, $\tilde{A}_{hH}$ are submatrices of A_{hh}, A_{Hh} and A_{hH} from (3.4), respectively.

This third way of implementation requires the use of the exact coarse grid subproblem that is sometimes difficult, see the next section. For this reason this third way of implementation will not be further discussed.

For the implementation of the FAC, AFAC and JFAC iterative methods, we need procedures for

- computation of the residual $r = b - Au$,
- restrictions of the residual $r_i = R_i r$, $i = 0, 1$,
- solution of the subproblems $A_i w_i = r_i$, $i = 0, 1$,
- prolongation and update $u \leftarrow u + I_i w_i$, $i = 0, 1$.

These operations are straightforward for the OB implementation. The restrictions then have the simplest form $R_H v = v_H$ and $R_h v = v_h$. One FAC iteration has the form

k-th FAC iteration

compute v_H^k: $A_H v_H^k = b_H - A_{HH} u_H^k - A_{Hh} u_h^k$
compute v_h^k: $A_h v_h^k = b_h - A_{hH}(u_H^k + v_H^k) - A_{hh} u_h^k$
$\left(u_H^{k+1}, u_h^{k+1}\right)^T = (u_H^k + v_H^k, u_h^k + v_h^k)^T$

A slight modification gives the FAC iteration in the typical Gauss-Seidel form

k-th FAC iteration

compute u_H^{k+1}: $A_H u_H^{k+1} = b_H - A_{Hh} u_h^k$
compute u_h^{k+1}: $A_h u_h^{k+1} = b_h - A_{hH} u_H^{k+1}$

For the NB implementation, we can use restrictions $R_h v = v_h$, $v = (\tilde{v}_H, v_h, v_{\mathcal{H}})^T$ and R_H given by

$$R_H v = R_{H\tilde{H}} \tilde{v}_H + R_{Hh} v_h + R_{H\mathcal{H}} v_{\mathcal{H}}.$$

In the case of fully compatible global and local grids, this restriction will be given by the interpolation $I_H \colon V_H \to V$,

$$I_H v_H = (I_{\tilde{H}} v_H, I_h v_H, I_{\mathcal{H}} v_H)^T,$$
$$R_{H\tilde{H}} = I_{\tilde{H}}^T, \quad R_{Hh} = I_h^T \quad \text{and} \quad R_{H\mathcal{H}} = I_{\mathcal{H}}^T.$$

In the case of not fully compatible global and local grids, we can in principle use any suitable interpolation and restriction. But in this case, we can lose some properties, which can deteriorate or destroy the convergence of FAC-type methods and the efficiency of the corresponding preconditioners. We will discuss these issues in the next section.

The FAC iteration in the NB implementation gets the form

k-th FAC iteration

1. compute $\tilde{r}_H^k = \tilde{b}_H - \tilde{A}_{HH} \tilde{u}_H^k - A_{H\mathcal{H}} u_{\mathcal{H}}^k$,
   $r_h^k = b_h - A_{hh} u_h^k - A_{h\mathcal{H}} u_{\mathcal{H}}^k$,
   $r_{\mathcal{H}}^k = b_{\mathcal{H}} - A_{\mathcal{H}H} \tilde{u}_H^k - A_{\mathcal{H}h} u_h^k - A_{\mathcal{H}\mathcal{H}} u_{\mathcal{H}}^k$,
2. solve $A_H v_H^k = R_{H\tilde{H}} \tilde{r}_H^k + R_{Hh} r_h^k + R_{H\mathcal{H}} r_{\mathcal{H}}^k$,
3. $u^{k+1/2} = u^k + I_H v_H^k = \big(\tilde{u}_H^k + I_{\tilde{H}} v_H^k, u_h^k + I_h v_H^k, u_{\mathcal{H}}^k + I_{\mathcal{H}} v_H^k\big)^T$
4. compute $r_h^{k+1/2} = b_h - A_{hh} u_h^{k+1/2} - A_{h\mathcal{H}} u_{\mathcal{H}}^{k+1/2}$
5. solve $A_h v_h^k = r_h^{k+1/2}$
6. put $u^{k+1} = \big(\tilde{u}_H^{k+1/2}, u_h^{k+1/2} + v_h^k, u_{\mathcal{H}}^{k+1/2}\big)^T$

Steps 4 and 5 can be rewritten as

$$A_h u_h^{k+1} = b_h - A_{h\mathcal{H}} u_{\mathcal{H}}^{k+1/2} \tag{3.10}$$

where the modification of b_h corresponds to incorporation of nonhomogeneous Dirichlet's boundary conditions given by $u_{\mathcal{H}}^{k+1/2} = u_{\mathcal{H}}^{k+1}$ on $\partial\Omega_R \setminus \partial\Omega$. This modification is called the delayed correction scheme in [14].

Lemma 3.1. *Let us consider the k-th FAC iteration ($k \geqslant 1$) in the NB implementation and assume that the subproblems are solved exactly and $\tilde{r}_H^0 = 0$. Then*

$$r_h^k = 0 \quad \textit{and} \quad \tilde{r}_H^k = 0.$$

P r o o f. We have

$$r_h^k = b_h - A_{hh} u_h^k - A_{h\mathcal{H}} u_{\mathcal{H}}^k = b_h - A_h u_h^k - A_{h\mathcal{H}} u_{\mathcal{H}}^{k-1/2} = 0$$

by virtue of (3.10). Further, we can prove $\tilde{r}_H^k = 0$ by induction:

$$\begin{aligned}
\tilde{r}_H^k &= \tilde{b}_H - \tilde{A}_{HH}\tilde{u}_H^k - A_{H\mathcal{H}}u_{\mathcal{H}}^k \\
&= \tilde{b}_H - \tilde{A}_{HH}\tilde{u}_H^{k-1} - A_{H\mathcal{H}}u_{\mathcal{H}}^{k-1} - \tilde{A}_{HH}\tilde{v}_H^{k-1} - A_{H\mathcal{H}}v_{\mathcal{H}}^{k-1} \\
&= \tilde{r}_H^{k-1} - R_{H\tilde{H}}(A_H v_H^{k-1}) \\
&= -R_{H\tilde{H}}(R_{H\tilde{H}}\tilde{r}_H^{k-1} + R_{Hh}r_h^{k-1} + R_{H\mathcal{H}}r_{\mathcal{H}}^k) \\
&= -R_{H\tilde{H}}\tilde{r}_H^{k-1} = 0.
\end{aligned}$$

The identity $\tilde{A}_{HH}\tilde{v}_H^{k-1} - A_{H\mathcal{H}}v_{\mathcal{H}}^{k-1} = R_{H\tilde{H}}(A_H v_H^{k-1})$ follows from the fact that the block $\tilde{A}_{HH}$ is contained in A_H and that $a(\Phi_i^H, \Phi_j^{\mathcal{H}}) = a(\Phi_i^H, \Phi_j^H)$ for any Φ_i^H corresponding to a coarse grid node in $\Omega \setminus \Omega_R$ and $\Phi_j^{\mathcal{H}}, \Phi_j^H$ corresponding to a coarse grid node on the interface $\partial\Omega_R \setminus \partial\Omega$. □

4. Inexact subproblems

The standard two-level composite grid FE method, its implementation and the convergence theory of FAC, AFAC and JFAC methods assume that

(i) both subproblem matrices are defined by the bilinear form a of the problem solved, see (3.1),

(ii) the global and local grids are fully compatible, which makes the representation (3.2) possible.

In practice, both of the above assumptions can be violated for some reasons, see Fig. 2 where we can see two examples of violation of the above conditions.

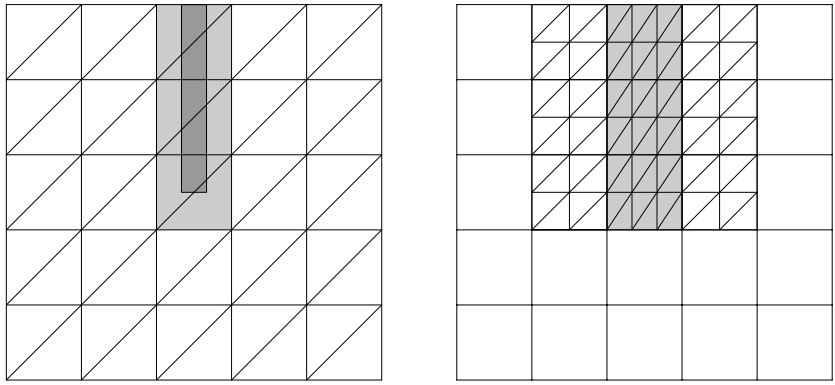

Figure 2. Violated coefficient variation (i) and violated compatibility of grids (ii) in the shaded area.

The first case arises when we do not permit variation of the problem coefficients within the coarse grid elements. This may be due to the exploited software, which assumes constant problem coefficients within the finite elements, or due to the hierarchical modelling procedure, which starts with a rough global model and continues with introducing a fine local grid and specifying the proper coefficients variation etc. In this manner the FAC method generalizes the simple submodelling technique, which can be found in the engineering literature, see e.g. [8]. Note that neglecting some coefficient variation may also lead to constant coefficient problems, which can be advantageously solved by some fast direct solver.

The second case, which is shown in Fig. 2, concerns the use of anisotropic grid refinement, which results in violation of the compatibility condition (ii). This case can occur very frequently, especially when solving 3D problems.

In both of the above cases we have to slightly modify the implementation of the composite grid FE method and the iterative methods.

Assume that the incompatibility occurs in $\Omega_0 \subset \Omega_R$, where Ω_0 can be composed from some of the coarse grid finite elements. Then the NB implementation of the composite grid FE method need not be changed, but in the iterative solution methods and preconditioners, we shall use the subproblem matrix $\tilde{A}_0$, which will be inexact in the sense that

$$\tilde{A}_0 \neq A_0 = R_0 A I_0. \tag{4.1}$$

We shall assume that $\tilde{A}_0$ remains symmetric positive definite.

In the OB implementation, we have to extract Ω_0 from the coarse grid FE space V_H. It means that V_H will be replaced by V_H^0,

$$V_H^0 = \{v \in \mathcal{V}\colon\ v \equiv 0 \ \text{ in } \ \Omega_0,\ \ v|_E \in P_1 \ \ \forall E \in \mathcal{T}_H(\Omega),\ \ E \subset \Omega \setminus \Omega_0\},$$

$V = V_H^0 + V_h$. Let A_H^0 be the stiffness matrix corresponding to V_H^0. Then A_H^0 must be used for implementation of the composite grid FE method, i.e. for computation of the residuals within the iterative methods and preconditioners. A_H^0 can be also used for computation of the corrections within the iterative methods and preconditioners. But practically, it will be more efficient to use FAC, AFAC etc. with the correction computed by means of the inexact subproblem matrix $\tilde{A}_0$ instead of A_H^0. Intuitively, FAC with A_H^0 is an overlapping Schwarz' method without the coarse grid acceleration, but $\tilde{A}_0$ provides the global subproblem, which is important for the efficiency of Schwarz' iterations. These effects will be demonstrated in the next section.

Now, we shall consider FAC, SFAC and JFAC methods and the corresponding preconditioners with the coarse grid corrections computed by means of $\tilde{A}_0$. If $\tilde{A}_0$ is

close to A_0, then the convergence of these methods can be proved as in the case of inexact solvers, [14], [5]. But if they are not very close, then the methods need not converge. For this reason, we shall introduce damping of the computed correction as a new tool for ensuring the convergence. If this damping concerns only the coarse grid correction, then we get modified FAC, SFAC and JFAC methods with the following iteration matrices:

$$\begin{aligned}\tilde{M}_{\mathrm{FAC}} &= (I - P_1)(I - \omega\tilde{P}_0),\\ \tilde{M}_{\mathrm{SFAC}} &= (I - P_1)(I - \omega\tilde{P}_0)(I - P_1),\\ \tilde{M}_{\mathrm{JFAC}} &= I - \frac{1}{2}(\omega\tilde{P}_0 + P_1),\end{aligned}$$

where $\tilde{P}_0 = I_0\tilde{A}_0^{-1}R_0A$, $P_1 = I_1A_1^{-1}R_1A$ and $\omega > 0$ is a damping parameter.

Theorem 4.1. *The modified FAC and SFAC methods are convergent provided the damping parameter is in the range* $0 < \omega < 2/\|\tilde{P}_0\|_A$.

P r o o f. First,

$$\varrho(\tilde{M}_{\mathrm{FAC}}) = \varrho\big((I - P_1)^2(I - \omega\tilde{P}_0)\big) = \varrho(\tilde{M}_{\mathrm{SFAC}}),$$

so that we can investigate only $\tilde{M}_{\mathrm{SFAC}}$, which is symmetric with respect to the inner product $\langle , \rangle_A$. As P_1 is an A-orthogonal projection $V \to V_1$, $\tilde{M}_{\mathrm{SFAC}}$ is zero on V_1. Moreover, $V_1, V_1^\perp$ are invariant subspaces of $\tilde{M}_{\mathrm{SFAC}}$ and therefore it is sufficient to investigate $\tilde{M}_{\mathrm{SFAC}}$ on $V_1^\perp$.

For further analysis, it is important that $\tilde{P}_0$ is positive definite on $V_1^\perp$. Obviously,

$$\langle\tilde{P}_0v, v\rangle_A = \langle AI_0\tilde{A}_0^{-1}R_0Av, v\rangle = \langle\tilde{A}_0^{-1}R_0Av, R_0Av\rangle \geqslant 0$$

because $\tilde{A}_0$ is positive definite. Moreover, $R_0Av \neq 0$ for $v \in V_1^\perp$, $v \neq 0$ because otherwise for any $w \in V$, $w = w_0 + w_1$, $w_i \in V_i$, we get

$$\langle v, w_1\rangle_A = \langle v, w_0\rangle_A + \langle v, w_1\rangle_A = \langle v, w_0\rangle_A = \langle Av, I_0w_0\rangle = \langle R_0Av, w_0\rangle = 0,$$

which contradicts $v \neq 0$. Thus $\tilde{P}_0$ is positive definite on $V_1^\perp$ and there is a constant $m_0 > 0$ such that

$$\langle\tilde{P}_0v, v\rangle_A \geqslant m_0\langle v, v\rangle_A \ \ \forall v \in V_1^\perp.$$

Thus,

$$\sigma(\tilde{M}_{\mathrm{SFAC}}) \subset \left\{ \frac{\langle \tilde{M}_{\mathrm{SFAC}} v, v\rangle_A}{\langle v, v\rangle_A} :\ v \in V_1^{\perp},\ \ v \neq 0 \right\} \cup \{0\} \tag{4.2}$$

$$= \left\{ \frac{\langle (I - \omega \tilde{P}_0) v, v\rangle_A}{\langle v, v\rangle_A} :\ v \in V_1^{\perp},\ \ v \neq 0 \right\} \cup \{0\}$$

$$\subset [1 - \omega \|\tilde{P}_0\|_A, 1 - \omega m_0] \cup \{0\}.$$

This yields $\varrho(\tilde{M}_{\mathrm{SFAC}}) = \varrho(\tilde{M}_{\mathrm{FAC}}) < 1$ for $\omega \in (0, 2/\|\tilde{P}_0\|_A)$. □

Theorem 4.2. *The modified JFAC method is convergent if the damping parameter is in the range* $0 < \omega < 3/\|\tilde{P}_0\|_A$.

P r o o f. As $\tilde{A}_0$, A_1 are symmetric positive definite, we have

$$\langle (\omega \tilde{P}_0 + P_1) v, v\rangle_A = \omega \langle \tilde{A}_0^{-1} R_0 A v, R_0 A v\rangle + \langle A_1^{-1} R_1 A v, R_1 A v\rangle \geqslant 0.$$

Moreover, for $v \neq 0$ we get $Av \neq 0$ and according to Lemma 2.1 at least one of the vectors $R_0 Av$, $R_1 Av$ must be nonzero. This shows that $\omega \tilde{P}_0 + P_1$ will be symmetric positive definite with respect to $\langle , \rangle_A$. The spectrum $\sigma(\omega \tilde{P}_0 + P_1) \subset [m(\omega), 1 + \omega \|\tilde{P}_0\|_A]$. Hence,

$$\sigma(\tilde{M}_{\mathrm{JFAC}}) \subset \Big[\frac{1}{2}(1 - \omega \|\tilde{P}_0\|_A), 1 - \frac{1}{2} m(\omega) \Big], \tag{4.3}$$

and

$$\varrho(\tilde{M}_{\mathrm{JFAC}}) \leqslant 1 \quad \text{for } \omega < 3/\|\tilde{P}_0\|_A.$$

□

Note that in the case of exact subproblems, the above theorems show that the damping is not necessary. Note also that $\|\tilde{P}_0\|_A$ is large and the damping is necessary if the inexact subproblem with $\tilde{A}_0$ corresponds to material which is substantially softer than the material of the exact subproblem. This will be illustrated in the next section.

Theorem 4.3. *Let* $\tilde{G}_{\mathrm{SFAC}}$ *and* $\tilde{G}_{\mathrm{JFAC}}$ *be the preconditioners which are defined by one SFAC and JFAC iteration with inexact subproblem matrix* $\tilde{A}_0$ *and without the use of damping. Then* $\tilde{G}_{\mathrm{SFAC}}$, $\tilde{G}_{\mathrm{JFAC}}$ *remain symmetric and positive definite.*

P r o o f. We have the relation

$$\tilde{G}_{\mathrm{SFAC}} A = I - \tilde{M}_{\mathrm{SFAC}}.$$

Using (4.2) we get that $\tilde{G}_{\mathrm{SFAC}}A$ has positive eigenvalues. As A is symmetric positive definite and $\tilde{G}_{\mathrm{SFAC}}$ is symmetric, $\tilde{G}_{\mathrm{SFAC}}$ must be also positive definite.

The proof for $\tilde{G}_{\mathrm{JFAC}}$ is the same, with use of (4.3) instead of (4.2). □

5. Numerical testing

For testing the performance of the iterative composite grid methods, we shall solve a 2D wall problem. This problem illustrated in Fig. 3 is formulated in a rectangular domain Ω of 37.2×31 metres. This domain contains a concrete wall of 1.2×15 metres with the elastic modulus $E = 31.5\,\mathrm{GPa}$, Poisson's ratio $\nu = 0.2$ and the density $\gamma = 2.5\,\mathrm{g/cm^3}$. The wall is surrounded by elastic clay with the elastic modulus $E = 19.88$ MPa, Poisson's ratio $\nu = 0.42$ and the density $\gamma = 1.85\,\mathrm{g/cm^3}$. The wall is loaded with the pressure $V = 1.5\,\mathrm{MPa}$ on the top of the wall. On the other sides of Ω, we assume zero normal displacements and zero tangential stresses.

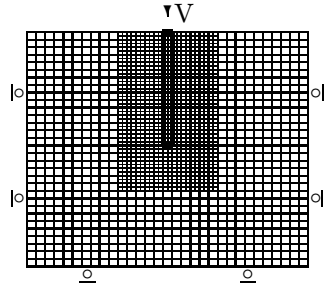

Figure 3. The test problem.

The global coarse grid uses 32×32 nodes with horizontal mesh size 1.2 m and vertical mesh size 1 m. The local fine grid uses halved grid sizes in the area which can be seen from Fig. 3. The fine mesh has 23×43 nodes. In both grids, the domain is divided into rectangles, which are subsequently divided into triangles.

The arising composite grid FE problem has been solved by all of the described iterative methods implemented in MATLAB. Table 1 shows the numbers of outer iterations which were required for obtaining the solution with the relative accuracy $\varepsilon = 10^{-6}$. In the first row, it is assumed that the subproblems are solved "exactly" by use of a direct solver (standard MATLAB procedure). In the second and third rows, the subproblems are assumed to be solved inexactly by an inner CG method controlled by the inner accuracy ε_i. Inexact solution, driven by the inner accuracy ε_i, was performed by an inner CG method.

	FAC	JFAC	AFAC	CG-SFAC	CG-JFAC	CG-AFAC
exact	10	44	22	6	13	12
$\varepsilon_i = 10^{-2}$	10	44	22	6	15	13
$\varepsilon_i = 10^{-1}$	13	43	22	10	29	27

Table 1. Numbers of outer iterations for various iterative composite grid methods, $\varepsilon = 10^{-6}$.

Further results concern the case of an inexact coarse grid problem. The OB formulation was used for testing the behaviour of FAC and CG-SFAC methods with the corrections computed with the aid of the matrices $A_H^0(0)$, $A_H^0(1)$, $A_H^0(2)$ and $\tilde{A}_H$. Here $A_H^0(k)$ are matrices which correspond to the coarse grid problem defined on $\Omega \setminus \Omega_0(k)$, where $\Omega_0(k)$ is the area of the wall enlarged by k layers of the neighbouring coarse grid elements. The matrices $A_H^0(k)$ are also used for the computation of the residuals. The matrix $\tilde{A}_H$ corresponds to the coarse grid problem on Ω with homogeneous soil material (the wall is replaced by soil). The corresponding results can be seen from Table 2.

	$A_H^0(0)$	$A_H^0(1)$	$A_H^0(2)$	$\tilde{A}_H$
FAC	36	46	61	14
CG-SFAC	10	10	13	7

Table 2. Numbers of iterations of FAC and CG-SFAC methods with corrections computed with the aid of various stiffness matrices, $\varepsilon = 10^{-6}$.

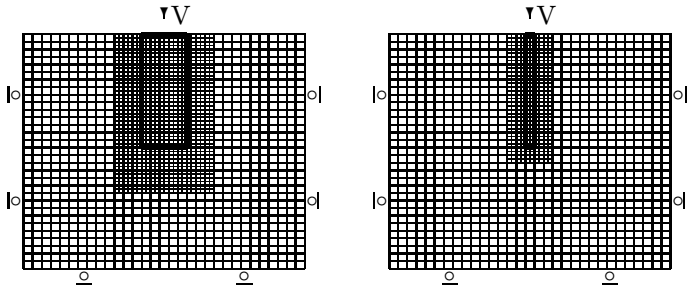

Figure 4. Wall problems with a) thicker wall and b) smaller refinement area.

The necessity of damping can be seen from Tab. 3, which shows numbers of iterations of FAC applied for solving the described test problem (Fig. 3), a problem with a thicker wall (Fig. 4a) and the former problem discretized with a smaller refinement region Ω_R (Fig. 4b). Here, we can see that FAC without damping diverges when solving the second and third problem. The problem from Fig. 4a was also used for testing the dependence of the number of iterations on the value of the damping parameter ω. The optimal value for FAC was found to be $\omega = 0.6$ giving 24 FAC iterations. For comparison, we show also the numbers of iterations for the CG-SFAC method for which $\omega = 1$ seems to be optimal.

	Problem Fig. 3		Problem Fig. 4a		Problem Fig. 4b	
	$\omega = 1$	$\omega = 1/2$	$\omega = 1$	$\omega = 1/2$	$\omega = 1$	$\omega = 1/2$
FAC	14	nt	∞	30	∞	31
CG-SFAC	6	nt	10	11	10	nt

Table 3. Number of iterations for damped FAC and CG-SFAC, nt=not tested, $\varepsilon = 10^{-6}$.

6. Concluding remarks

We have described the composite grid FE method which can be successfully applied to numerical solution of many boundary value problems provided the local grid refinement is required. The area of local grid refinement can be known a priori, as e.g. when solving multi-scale problems, or a posteriori, in dependence on the evaluation of error indicators for the solution on a primary coarse grid. The composite grid FE method is described together with natural iterative procedures for its solutions, which arise from the existing decomposition of the FE space.

We carefully describe several kinds of iterative procedures and their implementation, which is important for gaining additional benefits as a possibility of working with regular data structures corresponding to regular grids or parallelization of the computations.

We have also investigated the possibility of working with an inexact coarse grid subproblem, which is advantageous for hierarchical modelling which starts from a rough problem description on a coarse grid and continues with introducing local subgrids with more and more details of the problem analyzed.

We restrict our attention to two-level methods, but the extension to multi-level computations is mostly straightforward, see [12], [14], [17].

More details about the iterative methods including discussion of the use of variable and nonsymetric preconditioners, can be found in [5]. The inclusion of the iterative methods for the solution of composite grid FE problems into a broader class of space decomposition-subspace correction method is described e.g. in [9], [4].

Due to the restricted space, we consider only the solution of linear problems. For nonlinear problems, we can refer e.g. to [2] for applications in nonlinear elasticity and [3] for applications in plasticity.

Finally, we can mention that all the methods described are trivially parallelizable if the refinement of the global grid consists of several separated patches. But JFAC and AFAC methods and the corresponding preconditioners facilitate also "vertical" parallelization which can outweigh their worse convergence properties, see also [14].

References

[1] *O. Axelsson, V. A. Barker*: Finite Element Solution of Boundary Value Problems. Academic Press, Orlando, Florida, 1984.

[2] *R. Blaheta*: Iterative local refinement methods for nonlinear problems. HIPERGEOS report. IGAS Ostrava, 1998.

[3] *R. Blaheta*: Adaptive composite grid methods for problems of plasticity. Math. Comput. Simulation *50* (1999), 123–134.

[4] *R. Blaheta*: Space decomposition methods: displacement decomposition, composite grid finite elements and overlapping domain decomposition. In: Proceedings of the Conference Contemporary Mathematical Methods in Engineering (J. Doležalová, ed.). TU Ostrava, 2000, pp. 7–16.

[5] *R. Blaheta*: GPCG: CG method with general preconditioning and its applications. In: Proceedings of the Conference PRISM'01 (O. Axelsson, B. Polman, M. Neytcheva, eds.). KUN Nijmegen, The Netherlands, 2001, pp. 9–15.

[6] *D. Braess*: Finite Elements: Theory, Fast Solvers, and Applications in Solid Mechanics. Cambridge University Press, Cambridge UK, 1997.

[7] *J. H. Bramble, D. E. Ewing, J. E. Pasciak and A. H. Schatz*: A preconditioning technique for the efficient solution of problems with local grid refinement. Comput. Meth. Appl. Mech. Engrg. *67* (1988), 149–159.

[8] *R. D. Cook*: Finite Element Modeling for Stress Analysis. J. Wiley, New York, 1995.

[9] *T. F. Chan, T. P. Mathew*: Domain decomposition algorithms. Acta Numerica *3* (1994), 61–143.

[10] *L. Hart, S. F. McCormick*: Asynchronous multilevel adaptive methods for solving partial differential equations on multiprocessors: Basic ideas. Parallel Comput. *12* (1989), 131–144.

[11] *J. Mandel, S. F. McCormick*: Iterative solution of elliptic equations with refinement: the two-level case. In: Domain Decomposition Methods (T. F. Chan, R. Glowinski, J. Periaux and O. B. Widlund, eds.). SIAM, Philadelphia, 1989, pp. 81–92.

[12] *J. Mandel, S. F. McCormick*: Iterative solution of elliptic equations with refinement: the model multi-level case. In: Domain Decomposition Methods (T. F. Chan, R. Glowinski, J. Periaux and O. B. Widlund, eds.). SIAM, Philadelphia, 1989, pp. 93–102.

[13] *S. F. McCormick*: Fast adaptive grid (FAC) methods: theory for variational case. In: Defect Correction Methods: Theory and Applications (K. Böhmer, H. J. Stetter, eds.). Computing Supplementum, 5, Springer-Verlag, Wien, 1984, pp. 115–122.

[14] *S. F. McCormick*: Multilevel Adaptive Methods for Partial Differential Equations. SIAM, Philadelphia, 1989.

[15] *S. F. McCormick, D. Quinlan*: Asynchronous multilevel adaptive methods for solving partial differential equations on multiprocessors: Performance results. Parallel Comput. *12* (1989), 145–156.

[16] *P. S. Vassilevski, L. T. Zikatanov*: Local refinement solution of 3D elasticity equations. Project Report, COPERNICUS CP94-00820, 1995.

[17] *O. B. Widlund*: Optimal iterative refinement methods. In: Domain Decomposition Methods (T. F. Chan, R. Glowinski, J. Periaux and O. B. Widlund, eds.). SIAM, Philadelphia, 1989, pp. 114–125.

Authors' address: *R. Blaheta, P. Byczanski, R. Kohut*, Department of Applied Mathematics, Institute of Geonics, Academy of Sciences of the Czech Republic, Studentská 1768, 708 00 Ostrava, Czech Republic, e-mail: {blaheta, byczanski, kohut}@ugn.cas.cz.