Iterative local refinement methods for nonlinear problems.

Software for iterative local refinement methods.

R. Blaheta, R.Kohut [1]

Abstract. We consider iterative methods for local analysis and submodelling. The considered methods are based on the alternating Schwarz method and include the well known FAC (fast adaptive composite grid) method and a new PAS method. These methods are used also in the case of submodelling when some details of the solved problem are not visible on the coarse global grid. The iterative local refinement methods are applied to both linear and nonlinear problems, especially to problems of nonlinear elasticity. The performance of the considered methods is illustrated by solving test problems which arise from the foundation engineering with the aid of two developed programs, NFAC-1 and PAS-1.

Keywords: Linear and nonlinear elasticity, local refinement, submodelling, alternating Schwarz method, FAC and PAS methods, Newton-FAC method

---

[1] Institute of Geonics AS CR,
Studentská 1768, 708 00 Ostrava–Poruba, Czech Republic
Supported by Grant COPERNICUS CP 940820 of the European Commission and the grant OK – 157 of the Ministry of Education CR

Contents:

1 Introduction

In the finite element analysis of many engineering problems, we frequently meet the following two phenomena. Firstly, we require the FE grid to be locally refined to fit the highly nonlinear behaviour of the exact solution in some parts of the considered domain. Secondly, we need sufficiently fine FE grid to be able to model relative small details, e.g. small holes in a machine component or mine openings in large investigated domain.

In the case when the local grid refinement is required, we can either refine the global FE grid or use composite grids consisting from a global coarse grid and local fine grids (patches). The later approach is usually accompanied by an iterative solution method which include the solution of the global and local subproblems in each iteration. There are several advantages of such iterative approach to local analysis and submodelling:

- problem decomposition,
- possibility to use structured grids for global/local problems,
- adaptivity,
- possibility of parallelization.

Note that the iterative local refinement methods of the considered type firstly appeared in context of the multigrid methods, see e.g. [12], [10], [11] and the references herein. Namely, it was MLAT (multi–level adaptive technique) of A. Brandt, CS (correction scheme) and FAC (fast adaptive composite grid) of S. McCormick. Especially, the last method was further developed and analysed, see e.g. [8] and [9]. This method was afterwards given into the context of the overlapping domain decomposition (Schwarz) methods, see e.g. [12].

The main contribution of our paper can be seen in a systematic use of the Schwarz framework, in introducing the PAS method (similar to the FAC method) and in application of the both FAC and PAS methods to both linear and nonlinear problems. Moreover, we attempted to clear up the role of consistency between the global and local grids (grid consistency) and consistency between the global and local subproblems (consistency in material parameters). These topics seems to be of a great importance for the applications of the method.

Note also that our exposition utilizes only the singular composite grid problem without a special attention to the nodes on interface between global and local grids. Moreover, from the beginning, we do not require any relation between the global and local grids. Some relations in this respect will appear later as a tool for more efficient implementation of the methods.

This text does not include the proofs of some theorems, in this respect we refer to other literature or to the manuscript [5]. Also some topics were left to a next paper. It concerns mainly to preconditioners based on the FAC method, see e.g. [6], [1] and [17], and the questions of parallelization of the iterative local refinement methods.

The paper is organized as follows. The problems of linear and nonlinear elasticity are briefly described in Section 2. The considered iterative methods for local analysis will use the framework of the alternating Schwarz algorithm which is described in Section 3. Two specific algorithm of this kind, the FAC and PAS methods, are described in Section 4. In section 5, we shall describe the algebraic form of the composite grid problem and implementation of the FAC/PAS algorithms introduced in the previous Section. The applications of the FAC/PAS methods to nonlinear problems are described in Section 6. Here, we introduce also a different Newton - FAC/PAS when FAC/PAS are used for solving linearized equations. An important situation arises when we need to solve problem with some details which are not visible on the global coarse grid. This case is considered in Section 7. It is shown here that the PAS or a modified FAC with damping can be successfully used in this case. In Section 8, we shall describe NFAC-1 and PAS-1 software which implements the described methods. The paper is finished by description of a model problem of a pile foundation in clay with nonlinear behaviour of the soil and discussion of the results obtained by solving this problem by using the NFAC-1 and PAS-1 software. This software is described in more detail in Appendix 1.

Notation: Throughout this paper $\langle \quad , \quad \rangle$ denotes the eucledian inner product in R^n, $\langle u, v \rangle = u^T v$.

2 Linear and nonlinear elasticity

In this section, we shall describe the problems of elasticity with both linear and nonlinear material behaviour. For $\Omega \subset R^d, (d = 2, 3)$ such problems can be formulated with the aid of

- the displacement $u : \Omega \to R^d$,
- the strain vector $\varepsilon : \Omega \to S \subset R^{d^2}$,
- the stress vector $\sigma : \Omega \to S \subset R^{d^2}$.

For $x \in \Omega$ the vectors $\varepsilon(x), \sigma(x) \in S$ contains properly arranged components of the small strain tensor and the Cauchy stress tensor, respectively. For the details, see [14] or [3].

We shall assume the stress–strain relations which can be written in the form

$$\sigma = T(\varepsilon), \quad T : S \to S \tag{2.1}$$

or more precisely

$$\sigma(x) = T(x, \varepsilon(x)), \tag{2.2}$$

$$\sigma(x) = 3k(x, \hat{\varepsilon}(x)) \operatorname{vol}(\varepsilon(x)) + 2\mu(x, \bar{\varepsilon}(x)) \operatorname{dev}(\varepsilon(x)) \tag{2.3}$$

where k is the bulk modulus and μ is the shear modulus and

$$\hat{\varepsilon}(x) = \| \operatorname{vol}(\varepsilon(x)) \|^2, \quad \bar{\varepsilon}(x)) = \| \operatorname{dev}(\varepsilon(x)) \|^2 . \tag{2.4}$$

Here $\| \, . \, \|$ is the eucledian norm in R^{d^2} and for any $\eta = (\eta_{ij}) \in S$,

$$\operatorname{vol}(\eta) = (\hat{\eta}_{ij}), \hat{\eta}_{ij} = \frac{1}{d} (\eta_{11} + \ldots + \eta_{dd}) \delta_{ij}, \tag{2.5}$$

$$\operatorname{dev}(\eta) = \eta - \operatorname{vol}(\eta). \tag{2.6}$$

Above, δ_{ij} denotes the Kronecker's symbol.

Note that the quantity $\bar{\varepsilon}$ is in a simple relation with more frequently used intensity of shear strain ε_i, $\bar{\varepsilon} = 2\varepsilon_i^2$, see e.g. [14].

In the case of linear elasticity, the bulk modulus k and the shear modulus μ do not depend on the strains ε.

The *boundary value problem* of elasticity can be now written in the form

$$\sum_{i=1}^{d} \frac{\partial \sigma_{ij}}{\partial x_j} + f_i = 0 \qquad \text{in} \quad \Omega \subset R^d \quad (i = 1, \ldots, d) \tag{2.7}$$

$$\sigma(x) = T(x, \varepsilon(u(x))) \qquad \forall x \in \Omega \tag{2.8}$$

$$u = 0 \qquad \text{on} \quad \Gamma_0 \subset \partial\Omega\,, \quad \text{meas}\,(\Gamma_0) > 0\,, \tag{2.9}$$

$$\sum_{j=1}^{d} \sigma_{ij} \nu_j = \tau_i \qquad \text{on} \quad \Gamma_1 \subset \partial\Omega \quad (i = 1, \ldots, d)\,. \tag{2.10}$$

This boundary value problem can be formulated variationally in the space $U = \{v \in [H^1(\Omega)]^d, v = 0 \quad \text{on}\, \Gamma_0\}$ equipped with the following inner product and norm

$$(u, v)_1 = (u_1, v_1)_1 + \ldots + (u_d, v_d)_1, \quad \| v \|_1 = (v, v)_1^{1/2}$$

where $(\,,)_1$ is the standard inner product in $H^1(\Omega)$. For the details, see [14] and [3].

In the sequel, we shall assume that the material functions $k = k(x, s)$ and $\mu = \mu(x, s)$ are continuously differentiable with respect to the variable s and that the following conditions hold for any $(x, s) \in \bar{\Omega} \times \langle 0, \infty)$:

$$0 < k_0 \le k(x, s) \le k_1, \tag{2.11}$$

$$0 < \mu_0 \le \mu(x, s) < \frac{3}{2} k(x, s) \le \mu_1, \tag{2.12}$$

$$0 < k_0' \le \frac{\partial}{\partial s} \left[k(x, s^2) s \right] \le k_1', \tag{2.13}$$

$$0 < \mu_0' \le \frac{\partial}{\partial s} \left[\mu(x, s^2) s \right] \le \mu_1'. \tag{2.14}$$

Under these assumptions, we can prove that the problem of nonlinear elasticity is potential with monotone, Lipschitz continuous and differentiable nonlinear operator, see the following Theorem. For the details see [14], [13] and [3].

Theorem 2.1 *Let (2.11)–(2.14) hold.*

Then there is a functional $J : U \to R^1$ *such that*

- J *is differentiable with gradient* $J'(u)$,
- *if* $(,)_*$ *is the duality pairing, then*

$$\left(J'(u),\, v\right)_* \;=\; a(u;\, u,\, v) - b(v) \tag{2.15}$$

  where

$$a(w;\, u,\, v) \;=\; \int\limits_\Omega \langle T\,(\varepsilon(w))\;\varepsilon(u),\, \varepsilon(v)\rangle\, dx, \tag{2.16}$$

$$b(v) \;=\; \int\limits_\Omega f^T v dx + \int\limits_{\Gamma_1} \tau^T v\, ds, \tag{2.17}$$

- *there exists also the second derivative of* J,

$$\left(J''(w)u,\, v\right)_* \;=\; a'(w;\, u,\, v) \;=\; \int\limits_\Omega \langle T'\,(\varepsilon(w))\;\varepsilon(u),\, \varepsilon(v)\rangle\, dx \tag{2.18}$$

  where T' *is the derivative of the constitutive operator,*

- J *is strictly convex, especially there is a constant* $\alpha > 0$ *such that*

$$J(u) - J(v) - \left(J'(v),\, u - v\right)_* \;\geq\; \alpha \parallel u - v \parallel^2 \tag{2.19}$$

- *the gradient of* J *is Lipschitz continuous, i.e. there is a constant* K

$$\parallel J'(u) - J'(v) \parallel \leq\; K \parallel u - v \parallel \tag{2.20}$$

- *there is a unique weak solution* $u \in U$ *of the above elasticity problem,*

$$u \;=\; \operatorname{argmin}\{J(v) : v \in U\}. \tag{2.21}$$

Note that (2.19) simply follows from the monotonicity of J'.

A special case is represented by the problem of *linear elasticity* when $a(w;\, u,\, v)$ does not depends on w, i.e.

$$a(u;\, u,\, v) \;=\; a(u,\, v) \tag{2.22}$$

where $a(u,\, v)$ is a symmetric bilinear form in U,

$$J(v) \;=\; \frac{1}{2} a(v,\, v) - b(v)\,. \tag{2.23}$$

Notation: If $\Omega_0 \subset \Omega$ then we shall denote the restriction of a to Ω_0 as a_{Ω_0},

$$a_{\Omega_0}(w;\, u,\, v) \;=\; \int\limits_{\Omega_0} \langle T\,(\varepsilon(w))\;\varepsilon(u),\, \varepsilon(v)\rangle\, dx, \tag{2.24}$$

3 Alternating Schwarz algorithm

Let U be a Hilbert space, V_1 and V_2 be two finite dimensional subspaces of U , $V \;=\; V_1 + V_2$ and $J : U \;\rightarrow\; R^1$ be differentiable functional.

Then the minimization problem:

$$\text{find} \quad u \in V : u \;=\; \operatorname{argmin}\{J(v) : v \in V\}$$

can be solved by the following *alternating Schwarz algorithm*:

Algorithm 1a

Let $u^0 \in V$ is given. Then

for $i \;=\; 0, 1, \ldots$ **until** convergence **do**

$$v_1 \;=\; \operatorname{argmin}\{J(u^i + v) : v \in V_1\} \tag{3.1}$$

$$u^{i+1/2} \;=\; u^i + v_1 \tag{3.2}$$

$$v_2 \;=\; \operatorname{argmin}\{J(u^{i+1/2} + v) : v \in V_2\} \tag{3.3}$$

$$u^{i+1} \;=\; u^{i+1/2} + v_2 \tag{3.4}$$

enddo

The minimization can be expressed as finding arguments in which the corresponding gradient is zero. This, together with the expression of the gradient in the form (2.15), gives

Algorithm 1b

Let $u^0 \in V$ is given. Then

for $i = 0, 1, \dots$ **until** convergence **do**

$$\begin{aligned}
\text{find}\, v_1 \in V_1 \quad : \quad & a(u^{\,i} + v_1; u^{\,i} + v_1, v) \;=\; b(v) \quad \forall v \in V_1 && (3.5)\\
& u^{\,i+1/2} \;=\; u^{\,i} + v_1 && (3.6)\\
\text{find}\, v_2 \in V_2 \quad : \quad & a(u^{i+1/2} + v_2; u^{i+1/2} + v_2, v) \;=\; b(v) \quad \forall v \in V_2 && (3.7)\\
& u^{\,i+1} \;=\; u^{\,i+1/2} + v_2 && (3.8)
\end{aligned}$$

enddo

Theorem 3.1 *Let (2.19) and (2.20) hold. Then the alternating Schwarz algorithm is well defined and produces the sequence* $\{u^{\,i}\}$ *which converges to* $u \in V : u \;=\; \operatorname{argmin}\{J(v) : v \in V\}$.

Theorem 3.2 *The* inexact alternating Schwarz algorithm *uses approximate corrections*

$$\tilde{v}_i \in V_i, \quad \| \, \tilde{v}_i - v_i \, \| \leq \eta \, \| \, v_i \, \| \tag{3.9}$$

where v_i *are the exact corrections.*

Such algorithm converges for $\eta < 1$ *in the linear case or for* $\eta < \sqrt{2\alpha/K}$ *in the nonlinear case.*

The proof of Theorem 3.1 can be found in [15], the proof of Theorem 3.2 arises as a modification of the precedent proof, see also [5]. We can also prove a geometrical convergence of the methods, see e.g. [8], [15] and [1].

4 FAC and PAS methods

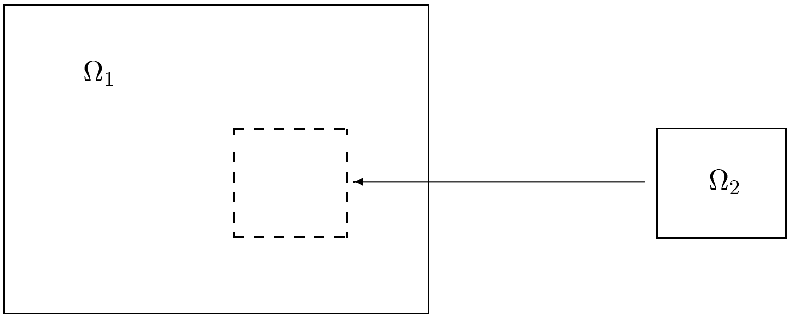

Figure 4.1: The domains Ω_1 and Ω_2 for the FAC method.

Let $\Omega_1 = \Omega$ and Ω_2 be a part of the domain Ω where a more accurate analysis is required, see Fig.4.1. Let T_1, T_2 be FE divisions of Ω_1, Ω_2 and let P be a set of given (polynomial) functions. Let us introduce two FE spaces

$$V_1 = \{v \in C(\bar{\Omega}) \cap U : v/E \in P \quad \forall E \in T_1\}$$

$$V_2 = \{v \in C(\bar{\Omega}) \cap U : v = 0 \text{ in } \Omega \backslash \Omega_2 \,,\, v/E \in P \quad \forall E \in T_2\}.$$

Then *FAC (fast adaptive composite grid method)* is the alternating Schwarz algorithm for finding the approximate solution

$$u = u_V = \min\{J(v) : v \in V = V_1 + V_2\}$$

where V_1 and V_2 are the FE spaces defined above. With respect to Theorems 3.1 and 3.2, the FAC method is convergent for both linear and nonlinear elasticity problems. Under suitable assumptions on V_1 and V_2, we are also able to prove mesh independent geometrical convergence of the method, see [8] and [1].

Let the domain Ω be divided into two overlapping subdomains Ω_1 and Ω_2 where Ω_2 is a part of the domain Ω where a more accurate analysis is

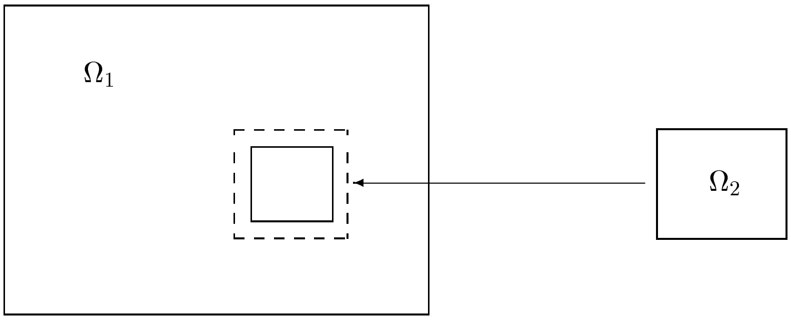

Figure 4.2: The domains Ω_1 and Ω_2 for the PAC method.

required, see Fig. 4.2. Let T_1, T_2 be FE divisions of Ω_1, Ω_2 and let P be a set of given (polynomial) functions. Let us introduce two FE spaces

$$V_1 = \{v \in C(\bar{\Omega}) \cap U : \quad v = 0 \quad \text{in}\, \Omega \setminus \Omega_1\,, \quad v/E \in P \quad \forall E \in T_1\}$$

$$V_2 = \{v \in C(\bar{\Omega}) \cap U : \quad v = 0 \quad \text{in}\, \Omega \setminus \Omega_2\,, \quad v/E \in P \quad \forall E \in T_2\}$$

Then *PAS (patched alternating Schwarz method)* is the alternating Schwarz algorithm for finding the approximate solution

$$u = u_v = \min\{J(v) : v \in V = V_1 + V_2\}$$

where V_1 and V_2 are the FE spaces defined above. With respect to Theorems 3.1 and 3.2, the PAS method is again convergent for both linear and nonlinear elasticity problems.

Note that in the case of $V_2^{FAC} = V_2^{PAS}$, $V^{FAC} = V^{PAS}$, we can expect that FAC will be a little faster.

5 FAC/PAS methods for linear problems

Algebraic formulation. For $k = 1, 2$, let $\{\phi_j^k : j = 1, \ldots, n_k\}$ be an arbitrary basis of V_k and $I_k : R_k \equiv R^{n_k} \to V_k$ be the corresponding isomorphism. Then any $v = v_1 + v_2,\, v_1 \in V_1,\, v_2 \in V_2$ can be represented by $\mathbf{v} = (\mathbf{v}_1, \mathbf{v}_2) \in R_1 \times R_2$, $v_k = I_k \mathbf{v}_k$.

For an algebraic version of the alternating Schwarz algorithm, we shall introduce matrices $A_{11}, A_{22}, A_{12}, A_{21}$ and vectors $\mathbf{b}_1 \in R_1$ and $\mathbf{b}_2 \in R_2$ by the identities

$$\langle A_{ij} \mathbf{u}_j, \mathbf{v}_i \rangle = a(I_j \mathbf{u}_j, I_i \mathbf{v}_i) \qquad \forall \mathbf{u}_i \in R_i,\, \mathbf{v}_j \in R_j \tag{5.1}$$

and

$$\langle \mathbf{b}_i, \mathbf{v}_i \rangle = b(I_i \mathbf{v}_i) \qquad \forall \mathbf{v}_i \in V_i\,. \tag{5.2}$$

This enables to express the conditions (3.5) and (3.7) from Algorithm 1b algebraically and rewrite this algorithm as

Algorithm 1c

Let $\mathbf{u}^0 = (\mathbf{u}_1^0, \mathbf{u}_2^0)$ is given. Then

for $i = 0, 1, 2, ...$ **until** convergence **do**

$$\text{find}\, \mathbf{v}_1 \in R_1 \;:\; A_{11} \mathbf{v}_1 = \mathbf{b}_1 - A_{11} \mathbf{u}_1^i - A_{12} \mathbf{u}_2^i = \mathbf{r}_1^i \tag{5.3}$$

$$\text{find}\, \mathbf{v}_2 \in R_2 \;:\; A_{22} \mathbf{v}_2 = \mathbf{b}_2 - A_{22} \mathbf{u}_2^i - A_{21} (\mathbf{u}_1^i + \mathbf{v}_1) = \mathbf{r}_2^i \tag{5.4}$$

$$\mathbf{u}_1^{i+1} = (\mathbf{u}_1^i + \mathbf{v}_1^i,\, \mathbf{u}_2^i + \mathbf{v}_2) \tag{5.5}$$

enddo

Note that A_{11}, A_{22} will be two SPD matrices, so that Algorithm 1c is well defined. Moreover, Algorithm 1c represents the block Gauss–Seidel iterative method applied to the linear system

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \end{bmatrix} \tag{5.6}$$

According to the results of Section 3, the iterations $\mathbf{u}^i$ converge to the solution of the above system. For checking the convergence in Algorithm 1c, we can use the following stopping criteria

$$\| \mathbf{v}_k \| \leq \varepsilon_k \| \mathbf{u}_k^i \| , \quad k = 1,2$$

or e.g.

$$\sqrt{\| \mathbf{r}_1 \|^2 + \| \mathbf{r}_2 \|^2} \leq \varepsilon \sqrt{\| \mathbf{b}_1 \|^2 + \| \mathbf{b}_2 \|^2}$$

The later conditions is simply the relative residual condition related to (5.6).

FAC/PAS methods with consistent meshes.

For the FAC/PAS methods, V_1 and V_2 are two FE spaces given by the FE divisions T_1 and T_2 of the domains Ω_1 and Ω_2, respectively. In this case, we shall assume that $\{\phi_j^k\}$ are the FE bases of V_k, $k = 1,2$. Therefore, the matrices A_{11} and A_{22} will be the standard stiffness matrices for the spaces V_1 and V_2. The other two matrices, are again matrices which can be assembled from local contributions, but now from the contributions of the intersections $E_1 \cap E_2$, $E_1 \in T_1$ and $E_2 \in T_2$.

So far, we have not assumed any relation between the bases in V_1 and V_2, i.e. any relation between the global and local grids. But the assembling of A_{12} and A_{21} can be simplified if the divisions T_1 and T_2 are *consistent.* This consistency means that

- $\Omega_0 \equiv \Omega_1 \cap \Omega_2$ can be decomposed into the elements from T_1,
- any $E \in T_1$, $E \subset \bar{\Omega}_2$ can be decomposed into the elements of T_2.

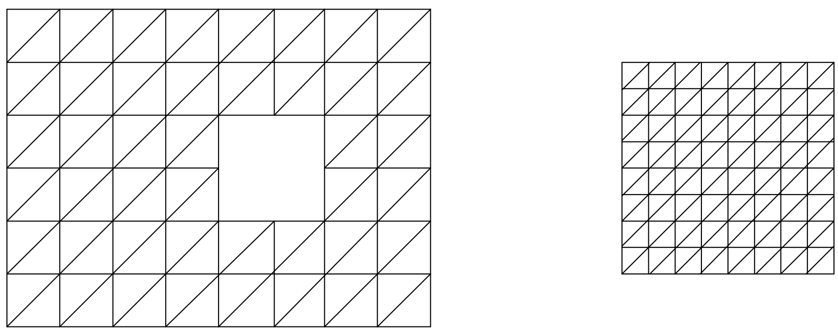

Figure 5.1: The PAS method with consistent meshes

In other words, we shall speak about FAC/PAS *methods with consistent meshes* , see Fig. 5.1. In the case of consistent meshes, we shall introduce an auxiliary space

$$V_0 \;=\; \{v \in C(\bar{\Omega}_0) \,:\, v/E \in P \quad \forall E \in T_2\,,\, E \subset \bar{\Omega}_0\} \tag{5.7}$$

Note that for the FAC method, the space V_0 is simply extension of V_2 corresponding to Neumann's type boundary conditions on the inner boundary $(\partial\Omega_2) \cap \Omega_1$.

In V_0 , we can choose a FE basis $\{\phi_j^0 \,:\, j \,=\, 1, \ldots, n_0\}$ which practically contains many elements from $\{\phi_j^2\}$. We can also define the corresponding isomorphism $I_0 \,:\, R_0 \,\equiv\, R^{n_0} \,\to\, V_0$ and two operators $I_1^0 \,:\, R_1 \,\to\, R_0$ and $I_2^0 \,:\, R_2 \,\to\, R_0$ which will be defined by the identities

$$I_1\mathbf{v}_1 \;=\; I_0\, I_1^0\, \mathbf{v}_1 \quad \text{in} \quad \bar{\Omega}_0 \qquad \forall \mathbf{v}_1 \in V_1, \tag{5.8}$$

$$I_2\mathbf{v}_2 \;=\; I_0\, I_2^0\, \mathbf{v}_2 \quad \text{in} \quad \bar{\Omega}_0 \qquad \forall \mathbf{v}_2 \in V_2 \tag{5.9}$$

We shall also need an auxiliary stiffness matrix corresponding to V_0 ,

$$\langle A_0\mathbf{u}_0,\, \mathbf{v}_0\rangle \;=\; a_{\Omega_0}(I_0\mathbf{u}_0,\, I\mathbf{v}_0) \qquad \forall \mathbf{u}_0,\, \mathbf{v}_0 \in R_0. \tag{5.10}$$

Then

$$A_{12} \;=\; \left(I_1^0\right)^T A_0\, I_2^0 \quad \text{and} \quad A_{21} \;=\; \left(I_2^0\right)^T A_0\, I_1^0\,. \tag{5.11}$$

The above relations immediately follows from the definitions, e.g.

$$\begin{aligned} \langle\, A_{12}\mathbf{u}_2,\, \mathbf{v}_1\rangle \;&=\; a\,(I_2\,\mathbf{u}_2,\, I_1\,\mathbf{v}_1) \;=\; a_{\Omega_0}\left(I_0 I_2^0\mathbf{u}_2,\, I_0 I_1^0\mathbf{v}_1\right) \\ &=\; \langle A_0\, I_2^0\,\mathbf{u}_2,\, I_1^0\mathbf{v}_1\rangle \;=\; \langle\left(I_1^0\right)^T A_0\, I_2^0\,\mathbf{u}_2,\, \mathbf{v}_1\rangle\,. \end{aligned}$$

The formulas (5.11) can be used for computer implementation of the computation of residuals from Algorithm 1c. Note that this computation can be also organized in the element–by–element way.

PAS algorithm with boundary transfers.

Simple implementation of the PAS method can be based on "overlapping representation" of the functions from $V \,=\, V_1 \,+\, V_2$ and the following variant of the basic algorithm.

Algorithm 2a

Let $u^0 = (u^0_{\Omega_1}, u^0_{\Omega_2})$, $u^0_{\Omega_2} = u^0_{\Omega_1}$ on Γ_2 is given. Then

for $i = 0, 1, 2, ...$ **until** convergence **do**

$$\text{find } u^{i+1}_{\Omega_1} \in V(\Omega_1) \quad : \quad u^{i+1}_{\Omega_1} = u^i_{\Omega_2} \quad \text{on} \quad \Gamma_1,$$

$$a_{\Omega_1}\left(u^{i+1}_{\Omega_1}, v\right) = b(v) \quad \forall v \in V_0(\Omega_1) \tag{5.12}$$

$$\text{find } u^{i+1}_{\Omega_2} \in V(\Omega_2) \quad : \quad u^{i+1}_{\Omega_2} = u^{i+1}_{\Omega_1} \quad \text{on} \quad \Gamma_2,$$

$$a_{\Omega_2}\left(u^{i+1}_{\Omega_2}, v\right) = b(v) \quad \forall v \in V_0(\Omega_2) \tag{5.13}$$

enddo

Above,

$\Gamma_k = \partial\Omega_k \cap \Omega$,

$V(\Omega_k) = \{v \in C(\bar{\Omega}_k) \cap U : \quad v/E \in P \quad \forall E \in T_k\}$

$V_0(\Omega_k) = \{v \in V(\Omega_k) : v = 0 \qquad \text{on} \quad \Gamma_k\}$, $k = 1, 2$.

An advantage of Algorithm 2 is in the fact that it requires only transfers of the boundary values. Therefore, implementation of this algorithm is very easy with a very little addition to the subproblem solvers.

A natural stopping criterion for Algorithm 2 will be given by the conditions

$$\| u^{i+1}_{\Omega_k} - u^i_{\Omega_k} \| \le \varepsilon \| u^i_{\Omega_k} \| \quad (k = 1, 2). \tag{5.14}$$

To clear up the relation between Algorithms 1 and 2, we shall need the following definition:

The FE divisions (meshes) T_k in Ω_k $(k = 1, 2)$ will be called *superconsistent* if there is a set $T_0 \subset T_1 \cap T_2$ such that $\Omega_0 = \Omega_1 \cap \Omega_2$ can be divided into the elements from T_0. An example of superconsistent meshes can be seen from Fig. 5.2.

Theorem 5.1 *Let us consider the PAS method with superconsistent meshes. Then the iterations of Algorithm 1 can be derived from the iterations of Algorithm 2.*

Proof. Algorithm 2 provides the functions

$$u^0_{\Omega_1}, u^0_{\Omega_2}, ..., u^i_{\Omega_1}, u^i_{\Omega_2}, ...$$

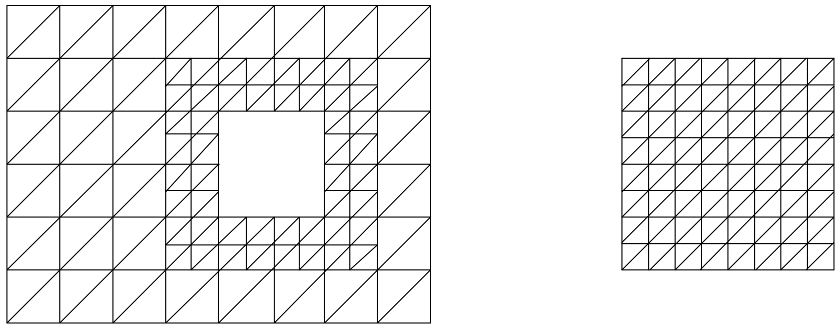

Figure 5.2: The PAS method with superconsistent meshes.

Let us define

$$u^{i} = \left\{ u^{i}_{\Omega_1} \text{ in } \bar{\Omega}_1 \backslash \Omega_2 \,;\, u^{i}_{\Omega_2} \text{ in } \Omega_1 \cap \Omega_2 \,;\, u^{i}_{\Omega_2} \text{ in } \bar{\Omega}_2 \backslash \Omega_1 \right\} ,$$

$$u^{i+1/2} = \left\{ u^{i+1}_{\Omega_1} \text{ in } \bar{\Omega}_1 \backslash \Omega_2 \,;\, u^{i+1}_{\Omega_1} \text{ in } \Omega_1 \cap \Omega_2 \,;\, u^{i}_{\Omega_2} \text{ in } \bar{\Omega}_2 \backslash \Omega_1 \right\} ,$$

$$u^{i+1} = \left\{ u^{i+1}_{\Omega_1} \text{ in } \bar{\Omega}_1 \backslash \Omega_2 \,;\, u^{i+1}_{\Omega_2} \text{ in } \Omega_1 \cap \Omega_2 \,;\, u^{i+1}_{\Omega_2} \text{ in } \bar{\Omega}_2 \backslash \Omega_1 \right\} .$$

Then,
$u^{i} \in v$,
$u^{i+1/2} = u^{i} + v_1$, where with respect to superconsistent meshes $v_1 \in V_1$,
$u^{i+1} = u^{i+1/2} + v_2$, where $v_2 \in V_2$.

Moreover,

$$\forall v \in V_1 \ : \ a(u^{i+1/2}, v) = a_{\Omega_1}(u^{i+1/2}, v) = a_{\Omega_1}(u^{i+1}_{\Omega_1}, v) = b_{\Omega_1}(v) = b(v) ,$$

$$\forall v \in V_2 \ : \ a(u^{i+1}, v) = a_{\Omega_2}(u^{i+1}, v) = a_{\Omega_2}(u^{i+1}_{\Omega_2}, v) = b_{\Omega_2}(v) = b(v) .$$

It remains to clear up what happens in the case of PAS algorithm with meshes which are consistent but not superconsistent. Assume that the mesh in Ω_1 can be refined to obtain superconsistent meshes (see at Fig. 5.1 and 5.2 for an example). Then Algorithm 2 can be viewed as an inexact version of the algorithm working with the refined superconsistent meshes.

6 FAC/PAS methods for nonlinear problems

For an algebraic formulation of Algorithm 1b, we use the same bases for the subspaces V_1 and V_2 and the same isomorphisms as in Section 5. The conditions (3.5) and (3.7) can be then rewritten to

$$\begin{aligned} & a(u^i + v_1,\, u^i + v_1,\, w_1) \\ = \;& a(I_1 \mathbf{u}_1^i + I_2 \mathbf{u}_2^i + I_1 \mathbf{v}_1\,; I_1 \mathbf{u}_1^i + I_2 \mathbf{u}_2^i + I_1 \mathbf{v}_1\,,\,, I_1 \mathbf{w}_1) \\ = \;& \langle A_{11}(\mathbf{u}_1^i + \mathbf{v}_1,\, \mathbf{u}_2^i)(\mathbf{u}_1^i + \mathbf{v}_1\,),\, \mathbf{w}_1\rangle + \langle A_{12}(\mathbf{u}_1^i + \mathbf{v}_1, \mathbf{u}_2^i)\, \mathbf{u}_2^i,\, \mathbf{w}_1\rangle \\ = \;& b(w_1) \;=\; \langle b_1\,,\, \mathbf{w}_1\rangle \end{aligned}$$

and

$$\begin{aligned} & a(u^{i+1/2} + v_2,\, u^{i+1/2} + v_2,\, w_2) \\ = \;& a(I_1(\mathbf{u}_1^i + \mathbf{v}_1) + I_2(\mathbf{u}_2^i + \mathbf{v}_2)\,, I_1(\mathbf{u}_1^i + \mathbf{v}_1) + I_2(\mathbf{u}_2^i + \mathbf{v}_2)\,,\,, \mathbf{w}_2) \\ = \;& \langle A_{22}(\mathbf{u}_1^i + \mathbf{v}_1,\, \mathbf{u}_2^i + \mathbf{v}_2)(\mathbf{u}_2^i + \mathbf{v}_2)\,,\, \mathbf{w}_2\rangle \\ + \;& \langle A_{21}(\mathbf{u}_1^i + \mathbf{v}_1,\, \mathbf{u}_2^i + \mathbf{v}_2)(\mathbf{u}_1^i + \mathbf{v}_1),\, \mathbf{w}_2\rangle = b(w_2) \;=\; \langle b_2\,,\, \mathbf{w}_2\rangle\,. \end{aligned}$$

Here, the matrices $A_{ij}(\mathbf{w}_1,\, \mathbf{w}_2)$ are defined by

$$\langle A_{ij}(\mathbf{w}_1,\, \mathbf{w}_2)\, \mathbf{u}_j,\, \mathbf{v}_i\rangle \;=\; a(I_1\mathbf{w}_1 +\, I_2\mathbf{w}_2;\, I_j\mathbf{u}_j,\, I_i\mathbf{v}_i) \tag{6.15}$$

This leads to the following version of Algorithm 1.

Algorithm 1d

Let $\mathbf{u}^0 \;=\; (\mathbf{u}_1^0,\, \mathbf{u}_2^0\,)$ is given. Then

for $i \;=\; 0, 1, 2, ...$ **until** convergence **do**

$$\text{find}\,\mathbf{v}_1 \in R_1 \;:\; A_{11}(\mathbf{u}_1^i + \mathbf{v}_1,\, \mathbf{u}_2^i)(\mathbf{u}_1^i + \mathbf{v}_1\,) \tag{6.16}$$

$$= \mathbf{b}_1 - A_{12}(\mathbf{u}_1^i + \mathbf{v}_1,\, \mathbf{u}_2^i)\, \mathbf{u}_2^i \;=\; \mathbf{r}_1^i \tag{6.17}$$

$$\text{find}\,\mathbf{v}_2 \in R_2 \;:\; A_{22}(\mathbf{u}_1^i + \mathbf{v}_1,\, \mathbf{u}_2^i + \mathbf{v}_2)\,(\mathbf{u}_2^i + \mathbf{v}_2) \tag{6.18}$$

$$= \mathbf{b}_2 - A_{21}(\mathbf{u}_1^i + \mathbf{v}_1,\, \mathbf{u}_2^i + \mathbf{v}_2)\,(\mathbf{u}_1^i + \mathbf{v}_1) \;=\; \mathbf{r}_2^i \tag{6.19}$$

$$\mathbf{u}_1^{i+1} \;=\; (\mathbf{u}_1^i \;+\; \mathbf{v}_1^i,\, \mathbf{u}_2^i \;+\; \mathbf{v}_2) \tag{6.20}$$

enddo

An disadvantage of this algorithm is in the fact that the unknowns $\mathbf{v}_1$, $\mathbf{v}_2$ appear also in the terms with A_{12} and A_{22}. One possible simplification is in deleting $\mathbf{v}_1$ and $\mathbf{v}_2$ from these terms which resembles the secant modulus method. Another possibility is in the use of the PAS method with overlapping representation of the solution and boundary transfers, i.e. use of the following

Algorithm 2b

Let $u^0 = (u^0_{\Omega_1}, u^0_{\Omega_2})$, $u^0_{\Omega_2} = u^0_{\Omega_1}$ on Γ_2 is given. Then

for $i = 0, 1, 2, ...$ **until** convergence **do**

$$\begin{aligned} \text{find} \quad u^{i+1}_{\Omega_1} \in V(\Omega_1): \quad u^{i+1}_{\Omega_1} &= u^{i}_{\Omega_2} \quad \text{on} \quad \Gamma_1, \\ a_{\Omega_1}\left(u^{i+1}_{\Omega_1}; u^{i+1}_{\Omega_1}, v\right) &= b(v) \quad \forall v \in V_0(\Omega_1) \end{aligned} \tag{6.21}$$

$$\begin{aligned} \text{find} \quad u^{i+1}_{\Omega_2} \in V(\Omega_2): \quad u^{i+1}_{\Omega_2} &= u^{i+1}_{\Omega_1} \quad \text{on} \quad \Gamma_2, \\ a_{\Omega_2}\left(u^{i+1}_{\Omega_2}; u^{i+1}_{\Omega_2}, v\right) &= b(v) \quad \forall v \in V_0(\Omega_2) \end{aligned} \tag{6.22}$$

enddo

The PAS algorithm is also advantageous in the case when nonlinearity is restricted in some subregion $\Omega_n \subset \Omega_2$. The choice $\Omega_n \cap \Omega_1 = \emptyset$ then restricts the use of nonlinear solver only to the subproblem in Ω_2.

In case of restricted nonlinearity, we can also use the FAC method with the nonlinearity located in Ω_2 and assume linear behaviour in the whole Ω_1. This leads to FAC with inconsistent materials (coefficients) which will be discussed in the following section.

Newton – FAC/PAS methods

The nonlinear elasticity problem in $V = V_1 + V_2$ can be written as

$$\text{find} \quad u \in V \ : a(u;\, u,\, v) = b(v) \qquad \forall v \in V. \tag{6.23}$$

This problem is frequently solved by some Newton–type iterative method of the following form:

$$u^0 \in V \tag{6.24}$$

$$u^{i+1} = u^n + \omega\, d^n \tag{6.25}$$

where $\omega \in (0, 1\rangle$ is an damping factor and $d^n \in V$ is an approximate solution of the linearized equation

$$a_n(d^n, v) \;=\; b(v) - a(u^n;\, u^n, v) \quad \forall v \in V \tag{6.26}$$

where a_n is an SPD bilinear form. The usual choices are

$$a_n(w, v) \;=\; a(u^n;\, w, v) \quad \forall w, v \in V, \tag{6.27}$$

which gives the *secant modulus method* and

$$a_n(w, v) \;=\; a'(u^n; w, v) \quad \forall w, v \in V \tag{6.28}$$

where a' is defined in Section 2. This gives the *Newton method.*

The *Newton–FAC/PAS method* now arises if the linearized equations (6.26) are solved approximately by one or more iterations of the FAC/PAS method.

7 Inconsistent global and local subproblems

Frequently, we need to solve problem with some details which are not visible on the global coarse grid. Such situation can be seen from Figure 7.1. Here the black boxes denote subdomain $\Omega_m \subset \Omega_2$ where the (material) coefficients differ from the material coefficients in $\Omega \setminus \Omega_m$. With respect to the size of the coarse grid elements, this difference is not considered in the coarse global grid subproblem.

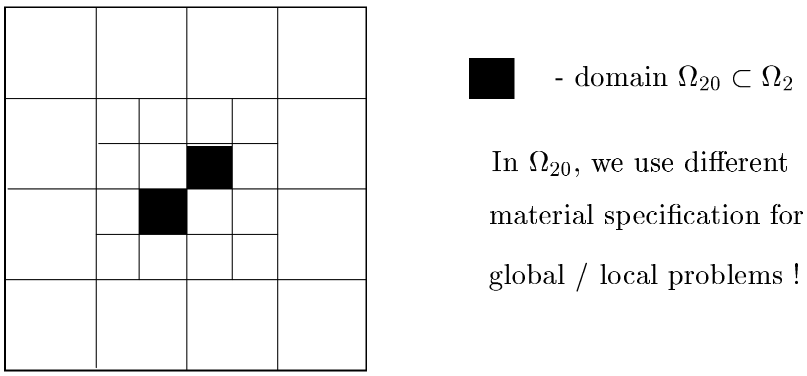

Figure 7.1: Problem with inconsistent subproblems

First approach, which is possible in the case of inconsistent subproblems is the use of the PAS method with the subdomain Ω_1 and Ω_2 chosen in such a way that $\Omega_1 \cap \Omega_m = \emptyset$.

Another approach is the use of modified FAC method. For this we have to perceive that for the global problem we can only assembly the matrix $\tilde{A}_{11}$ of the modified problem in Ω_1. On the other hand, we are able to compute the matrix by vector product with the matrix A_{11} corresponding to unmodified problem in Ω_1. This product $A_{11} w_1$ can be computed in the element–by–element way (also with the use of the prolongation / restriction operators).

This means that modified FAC will use

- Algorithm 1c or 1d,
- residuals $\mathbf{r}_1^i$ which are computed element–by –element or using the prolongation / restriction operators introduced in Section 5 for the case of consistent meshes,
- damping of the correction $\mathbf{v}_1$, by $\omega \in (0, 1 >$, i.e.

  $\mathbf{u}^{i+1/2} = \mathbf{u}^i + \omega \, \mathbf{v}_1^i$

This damping is necessary for guarantying the convergence, see [5].

8 Test problems

In this Section, we shall describe nonlinear elasticity problems which will be used for testing the performance of the iterative local refinement algorithms.

Figure 8.1 shows a single pile in a homogeneous clay soil which will be used as a test problem. The computational domain is a parallelepiped $\Omega = (0, 37.2) \times (0, 37.2) \times (0, 31)$ metres. The boundary conditions are chosen as follows: zero normal displacements and zero tangential stresses on all sides except the top one, Neumann boundary condition on the top side where the vertical load V = 1.5 MPa on the pile is applied. The volume force is formed by the weight of materials: $\gamma = 2.5 \ g/cm^3$ for the concrete pile and $\gamma = 1.8 \ g/cm^3$ for the soil.

The localization of the pile can be seen from the Figure 8.1, the dimensions of the pile are $1.2 \times 1.2 \times 15$ metres. The material of the pile is the concrete with the following mechanical characteristics: $E = 31500$ MPa, $\nu = 0.2$.

For the soil, we suppose nonlinear elastic behaviour which is described by the bulk modulus $k = 41.417$ and the shear modulus μ given by the following hyperbolic formula:

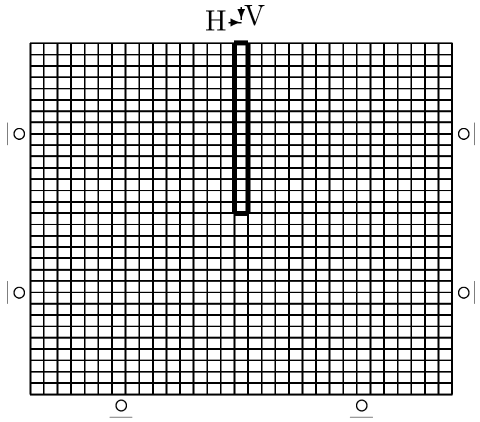

Figure 8.1: Single pile in a clay - cross section of the 3D domain.

$$\mu(\bar{\varepsilon}) = \frac{A}{B + \sqrt{\bar{\varepsilon}}} \quad [\mathrm{MPa}] \tag{8.1}$$

where $A = 0.0658$ and $B = 0.0094$. This hyperbolic relation was chosen as a good regression of the experimental data described in [7].

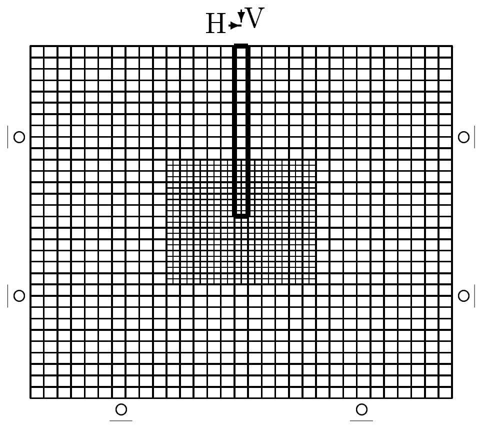

Figure 8.2: Test problem 1.

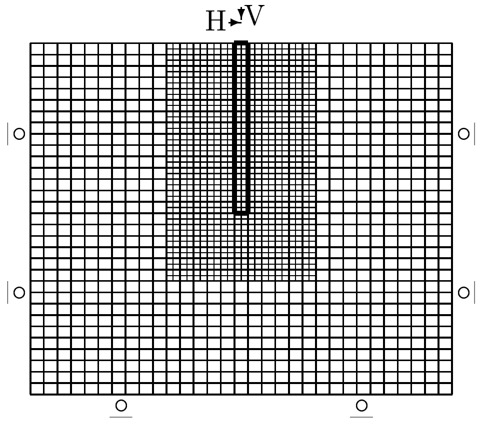

Figure 8.3: Test problem 2.

The value of μ for initial state ($\bar{\varepsilon} = 0$) correspond to the values $E = 19.88$ MPa, $\nu = 0.42$. Note that,

$$E = 3k\frac{\mu}{k + \frac{1}{3}\mu} \quad [MPa] \qquad \nu = \frac{k - \frac{2}{3}\mu}{2k + \frac{2}{3}\mu}$$

which implies that μ and E tend to zero and ν tends to $1/2$ if $\bar{\varepsilon}$ grows. The behaviour of μ can be seen from Figure 8.4. Note that the function (8.1) meets the assumptions formulated in Section 2 in any finite interval $0 \le \bar{\varepsilon} \le \bar{\varepsilon}_{max}$. On the other hand, the hyperbolic formula correctly works only if the existing stresses fulfils the restriction $\sqrt{\bar{\sigma}} \le 2A = 131.6$ kPa.

The problem is discretized by the finite element method with a regular rectangular grid. The discretization is given by the decomposition of the domain into eight node bricks which are consequently divided, each brick into 6 linear tetrahedral finite elements. The basic coarse grid has 32x32x32 = 32768 nodes.

For testing of the iterative local refinement methods, we shall specify two test problems which differs in the size of the refined subdomain, see Figures 8.2 and 8.3.

Test problem 1

The refined area (patch) is situated around the bottom part of the pile, see Figure 8.2. The fine mesh has 23x23x23 = 12167 nodes.

Test problem 2

The refined area (patch) from the previous test problem is prolongated up to the top surface, i.e. the whole pile is located in the patch, see Figure 8.3. The fine mesh has 23x23x43 = 22747 nodes.

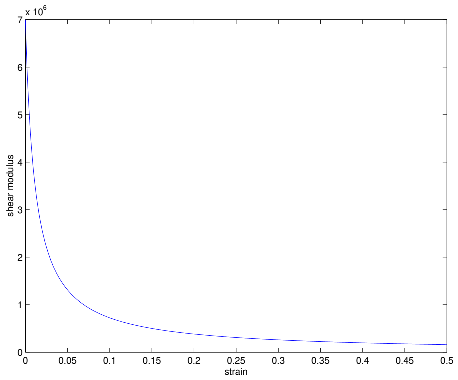

Figure 8.4: Shear modulus for the clay.

9 Numerical tests

The test problems, which were defined in the previous section, will be now used for testing two iterative local refinement methods, the Newton–FAC method and the PAS method. The both methods use the secant modulus iterations (6.27) for solving the arising nonlinear systems. The implementation of these methods in software moduli NFAC–1 and PAS–1 is described in Appendix 1. Note that the PAS method utilizes overlapping domains with overlap size equal to size of two coarse grid elements.

Initial guesses.

We shall consider some algorithmic variants given by the choice of initial guesses. For the Newton-FAC method, we can use the solution of the nonlinear problem on global coarse grid as initial guess. For the PAS method, we can again start from the solution of global coarse grid problem in the whole domain. Moreover, we can use the results from the previous PAS iteration as an initial guess for the solution of nonlinear problems in the subdomains during the next iteration.

Stopping criteria.
For the Newton–FAC method, we shall use the stopping criterion based on the condition

$$\| \mathbf{r}_1^i \|_\infty / \| \mathbf{b}_1 \|_\infty < 10^{-3} . \tag{9.2}$$

Hence, we do not consider the second residual component $\mathbf{r}_2^i$ and our condition is therefore a simplification of the stopping conditions which were discussed in the previous sections.

For the PAS method, we shall use the convergence test based on the change of the solution in the overlap $\Omega_{12} = \Omega_1 \cap \Omega_2$,

$$\frac{\| u^i - u^{i-1} \|_{\Omega_{12},\infty}}{\| u^i \|_{\Omega_{12},\infty}} < \varepsilon . \tag{9.3}$$

This stopping criterion can be easily implemented but unfortunately its use does not enable a clear comparison between the Newton–FAC and PAS methods. Therefore, we would like to implement also the residual criterion for the PAS method (which requires transfer of the displacement not only on the part of the boundary but also inside the subdomains).

The performance of the Newton–FAC method can be seen from Tables 1-5. Here, the 2nd and 5th columns display the numbers of inner PCG iterations, the 3rd and 6th columns show the norm of the difference in computed displacements which are in the 4th and 7th column related to the displacement computed in the first Newton–type iteration. The last column shows the norm of the first component of the residual which is used in the exploited stopping criterion (9.2).

The performance of the PAS method can be seen from Tables 6 and 7. Here, the 2nd and 4th columns contain number of the secant modulus iterations which were used for solving the nonlinear subproblems. The 3rd and 5th columns contain number of inner PCG iterations. The last column shows the value of the relative difference in the displacement which is used in the stopping criterion (9.3).

The displayed computer times concerns the execution of the software NFAC–1 and PAS–1 on IBM RS/6000 3CT computer.

	global grid			local grid			
IT	it	$\parallel du \parallel_\infty$	%	it	$\parallel du \parallel_\infty$	%	$\parallel \mathbf{r}_1^i \parallel_\infty$
1	94	0.16807	100	27	0.15576	100	0.25326
2	99	0.01430	8.508	30	0.01430	9.181	0.07173
3	89	0.00176	1.047	29	0.00176	1.048	0.02509
4	72	0.00040	0.236	29	0.00040	0.236	0.00961
5	68	0.00012	0.070	29	0.00012	0.070	0.00384
6	67	0.00004	0.023	28	0.00004	0.023	0.00161
7	67	0.00001	0.008	28	0.00001	0.008	0.00074

Table 1: N-FAC, test problem 1, no initial guess, $\parallel \mathbf{r}_1^i \parallel_\infty \leq 0.001$, inner PCG solver accuracy $\varepsilon_l in = 0.001$, CPU time 0:31:56, real time 0:33:05.

	global grid			local grid			
IT	it	$\parallel du \parallel_\infty$	%	it	$\parallel du \parallel_\infty$	%	$\parallel \mathbf{r}_1^i \parallel_\infty$
1	56	0.16924	100	6	0.15568	100	0.19810
2	44	0.01443	8.526	8	0.01444	9.250	0.05814
3	34	0.00179	1.058	8	0.00178	1.143	0.02054
4	34	0.00046	0.272	8	0.00045	0.289	0.01012
5	31	0.00014	0.083	7	0.00014	0.090	0.00892
6	34	0.00005	0.030	8	0.00005	0.032	0.00212
7	28	0.00002	0.011	5	0.00002	0.013	0.00075

Table 2: N-FAC, test problem 1, no initial guess , $\parallel \mathbf{r}_1^i \parallel_\infty \leq 0.001$, inner PCG solver accuracy $\varepsilon_l in = 0.1$, CPU time 0:19:10, real time 0:19:46.

	global grid			local grid			
IT	it	$\parallel du \parallel_\infty$	%	it	$\parallel du \parallel_\infty$	%	$\parallel \mathbf{r}_1^i \parallel_\infty$
1	–	0.17846	100	4	0.17003	100	0.08033
2	29	0.00136	0.762	8	0.00217	1.276	0.03047
3	32	0.00081	0.454	8	0.00081	0.476	0.01396
4	32	0.00026	0.146	8	0.00026	0.153	0.00456
5	33	0.00009	0.050	6	0.00009	0.053	0.00243
6	30	0.00003	0.017	7	0.00003	0.018	0.00078

Table 3: N-FAC, test problem 1, with initial guess, $\parallel \mathbf{r}_1^i \parallel_\infty \leq 0.001$, inner PCG solver accuracy $\varepsilon_l in = 0.1$, CPU time 0:13:31, real time 0:14:03.

	global grid			local grid			
IT	it	$\Vert du \Vert_\infty$	%	it	$\Vert du \Vert_\infty$	%	$\Vert \mathbf{r}_1^i \Vert_\infty$
1	56	0.16924	100	31	0.16787	100	0.11629
2	47	0.01264	7.469	38	0.01120	6.672	0.01578
3	41	0.00072	0.426	11	0.00073	0.432	0.00424
4	38	0.00021	0.124	43	0.00030	0.179	0.00124
5	29	0.00005	0.032	7	0.00006	0.035	0.00057

Table 4: N-FAC, test problem 2, no initial guess $\Vert \mathbf{r}_1^i \Vert_\infty \leq 0.001$, inner PCG solver accuracy $\varepsilon_l in = 0.1$, CPU time 0:19:56, real time 0:20:25.

	global grid			local grid			
IT	it	$\Vert du \Vert_\infty$	%	it	$\Vert du \Vert_\infty$	%	$\Vert \mathbf{r}_1^i \Vert_\infty$
1	–	0.17846	100	4	0.17846	100	0.04404
2	34	0.00137	0.768	11	0.00234	1.311	0.01351
3	33	0.00078	0.437	33	0.00108	0.605	0.00380
4	35	0.00019	0.106	8	0.00020	0.112	0.00142
5	38	0.00006	0.034	6	0.00006	0.034	0.00067

Table 5: N-FAC, test problem 2, with initial guess $\Vert \mathbf{r}_1^i \Vert_\infty \leq 0.001$, inner PCG solver accuracy $\varepsilon_l in = 0.1$, CPU time 0:14:31, real time 0:15:35.

PAS	it.SM	it.CG	it.SM	it.CG	$\frac{\|u_i - u_{i-1}\|_\infty}{\|u_i\|_\infty}$
1	2	154	2	28	-
2	1	2	1	6	0.004433
3	1	7	1	4	0.002657
4	1	3	1	5	0.001248
5	1	6	1	4	0.001075
6	1	3	1	5	0.000601
7	1	6	1	5	0.000563
8	1	5	1	5	0.000392
9	1	5	1	5	0.000321
10	1	5	1	5	0.000324
$\sum$	11	196	11	72	–

Table 6: PAS, test problem 1, with initial guess, CPU time 0:29:51, real time 0:31:10.

PAS	it.SM	it.CG	it.SM	it.CG	$\frac{\|u_i - u_{i-1}\|_\infty}{\|u_i\|_\infty}$
1	2	154	2	141	-
2	1	2	1	39	0.008023
3	1	3	1	46	0.004910
4	1	2	1	62	0.003197
5	1	3	1	40	0.002167
6	1	2	1	45	0.001724
7	1	6	1	37	0.001394
8	1	2	1	66	0.001151
9	1	3	1	46	0.000949
10	1	2	1	65	0.000783
$\sum$	11	179	12	587	

Table 7: PAS, test problem 2, with initial guess, CPU time 0:50:21, real time 0:51:35.

From the performed numerical tests, we can draw the following conclusions:

1. Comparing the results from Tables 1 and 2, we can see that a lower accuracy for the inner PCG solver is advantageous for the Newton–FAC method.

2. Comparing the results from Tables 2 and 3, we see that the initial guess further improves the performance of the Newton–FAC method.

3. Comparing the results of Tables 2 and 4 or Tables 3 and 5, we can conclude that larger patch improves the accuracy of one FAC iteration which induce here a lower number of the Newton iterations.

4. We can also see that the solution of linear problems for the local grid requires less number of inner PCG iterations. This can be explained by the pure Dirichlet boundary conditions which are used for the local subproblems.

5. The reported computations were performed in REAL*4 arithmetics. This accuracy seems to be a somewhat low. A further experiments in this respect will be done.

6. To be able to compare the Newton–FAC and PAS methods, we have to implement another stopping condition for the PAS method.

References

[1] O. Axelsson, P.S. Vassilevski: Iterative Methods Based on Overlapping Space Decompositions, manuscript, Nijmegen 1995

[2] R. Blaheta, R. Kohut: Efficient inexact Newton–like methods with application to problems of the deformation theory of plasticity, Appl. Math. 38 (1993), 411-427

[3] R. Blaheta: Elasticity with nonlinear behaviour of material, manuscript, 1996

[4] R. Blaheta: Multi-level Newton methods for nonlinear problems with application to elasticity, HIPERGEOS report, Institute of Geonics AS CR, Ostrava 1997

[5] R. Blaheta: Iterative local refinement methods, manuscript, 1998

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

[7] Asen Georgiev, Angel Baltov and Svetozar Margenov: HIPERGEOS Benchmark Problems Related to Bridge Engineering Applications, COPERNICUS CP 94-0820 Project Report, MOST-XXI, Sophia 1996

[8] Jan Mandel and Steve McCormick: Iterative Solutin of Elliptic Equations with Refinement: The Two–Level Case, in Domain Decomposition Methods, Edited by Tony F. Chan, Roland Glowinski, Jacques Periaux, Olof B. Widlund, SIAM Philadelphia 1988

[9] Jan Mandel and Steve McCormick: Iterative Solution of Elliptic Equations with Refinement: The Model Multi–Level Case, in Domain Decomposition Methods, Edited by Tony F. Chan, Roland Glowinski, Jacques Periaux, Olof B. Widlund, SIAM Philadelphia 1988

[10] S. McCormick: Fast Adaptive Composite Grid (FAC) Methods: Theory for the Variational Case, Computing, Suppl 5(1984), 115–121

[11] S. McCormick, J. Thomas: The Fast Adaptive Composite Grid Method for Elliptic Boundary Value Problems, Mathh. Comp. 46(1986), 439–456

[12] Steve McCormick: Multilevel Adaptive Methods for Partial Differential Equations, Frontiers in Applied Mathematics, Vol. 6, SIAM, Philadelphia 1989

[13] J. Nečas: Introduction to the theory of nonlinear elliptic equations, Teubner Texte zur Math., Band 52, Teubner, Leipzig 1983

[14] J. Nečas, I. Hlaváček: Mathematical theory of elastic and elasto–plastic bodies: An introduction, Elsevier, Amsterdam 1981

[15] P.L. Lions: On the Schwarz Alternating Method. I, 1^{st} Internat. Symp. on Domain Decomposition Methods for Partial Differential Equations, R. Glowinski, G.H. Golub, G.A. Meurant, J. Periaux eds. SIAM Philadelphia, 1988

[16] Barry F. Smith, Peter E. Bjorstad, and William D. Gropp: Domain Decomposition Parallel Multilevel Methods for Elliptic Partial Differential Equations, Cambridge University Press, 1996

[17] Panayot S. Vassilevski and Ludmil T. Zikatanov: Local refinement solution of 3D elasticity equations, Project Report, COPERNICUS CP94–00820