Multiscale computations with a combination of the $h$- and $p$-versions of the finite-element method

Roland Krause a,1, Ernst Rank b,*

a Weidlinger Associates Inc., 4410 El Camino Real, Los Altos, CA 44020, USA
b Lehrstuhl für Bauinformatik, Technische Universität München, Arcistr. 21, D-80290 München, Germany

Received 27 July 1998; received in revised form 25 April 2003; accepted 1 May 2003

Abstract

The numerical treatment of structural problems is often difficult if the global system behavior is affected by phenomena on different length scales. When simulating multiscale problems, it is essential to find a discretization that equally reflects all aspects of the problem with sufficient accuracy. The present paper is concerned with the formulation of a finite-element procedure which combines the $p$-method to resolve global features of a problem with an $h$-method to resolve local features. Thus the advantageous properties of both standard procedures can be optimally exploited. The implementation provides the further advantage that the numerical treatment of the problems on the different length scales can be done with independent finite-element discretizations. To solve the overall problem, an iteration scheme with an adaptive solution strategy is developed. Benchmark problems and an example with relevance to soil mechanics are presented.

© 2003 Elsevier B.V. All rights reserved.

Keywords: Multiscale finite-element method; $p$-Version; Adaptive $h$-version

1. Introduction

The combination of $h$- and $p$-extensions of the finite-element method for linear elliptic boundary value problems leads to exponential convergence rates of the error in energy norm if geometrical meshes are used where all singularities are sufficiently resolved. These meshes can automatically be constructed for certain two-dimensional problems.

The situation is more complicated for the class of multiscale problems, where two or more different length scales interact, as construction of a finite-element discretization equally reflecting the problem on all length scales is difficult. As an example, a highly complex geometry on the smallest scale can, with reasonable effort, only be resolved with an $h$-version finite-element mesh. On the other hand this might not be
accurate enough to correctly reflect the global behavior of the system and the interaction of the solutions on both scales. There are different known approaches for multiscale problems combining finite-element and boundary-element methods [12]. A further technique is the s-method proposed by Fish et al. [5–10]. A method using substructuring for the coupling of a laminated plate model and a three-dimensional model to investigate cracks in laminated materials is developed by Reddy and coworkers [24,25]. Another approach of decomposing and re-composing finite-element discretizations on different length scales is the mortar finite-element method recently investigated by Seshaiyer and Suri and [26]. The effects of microstructure on different scales on the macroscopic response of complex heterogeneous structures can be resolved with a homogenized dirichlet projection method proposed by Oden and Zohdi [17]. Finally, a combination of h- and p-extensions in a hierarchical domain decomposition (which will be abbreviated by hp-d method in the following) was first proposed in [19–22] and will be further investigated in this contribution.

The paper is organized as follows. In Section 2 the governing equations of the model problem and a finite-element approximation are presented. In Section 3 the basic idea of the hierarchical overlay of p- and h-version shape function spaces is explained. Following, in Section 4 the implementation of the method using a block iterative solution procedure for the resulting equation system is demonstrated. Further, an adaptive strategy enhancing the performance of the iteration is outlined. Various aspects of the method are discussed for two benchmark examples in Section 5. Finally an application of the method to a soil mechanics problem is presented in Section 6.

2. Governing equations

To set up our model problem, the equilibrium conditions of the three-dimensional continuum, described on a domain $\Omega$ in the $xy$-plane with thickness $t$, in the $z$-direction, are given by the system of partial differential equations

$$\begin{align*}
\frac{\partial \sigma_x}{\partial x} + \frac{\partial \tau_{xy}}{\partial y} + X &= 0, \\
\frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \sigma_y}{\partial y} + Y &= 0
\end{align*}$$

(1)

with stresses denoted in vectorial notation as $\sigma = [\sigma_x; \sigma_y; \tau_{xy}]^T$ and volumetric forces $f = [X; Y]^T$. On the boundary of the domain either displacements $u$ or tractions $T$ in normal and/or tangential direction can be prescribed as

$$u = 0 \text{ on } \Gamma = \hat{\Gamma}_u, \quad T = \hat{T} \text{ on } \Gamma = \hat{\Gamma}_T.$$  

(2)

We will assume a linear relation $\varepsilon = Lu$ between strains $\varepsilon = [\varepsilon_x; \varepsilon_y; \gamma_{xy}]^T$ and the displacement vector $u = [u_x(x,y); u_y(x,y)]^T$, the linear strain operator being given by

$$L = \begin{bmatrix}
\frac{\partial}{\partial x} & 0 & \frac{\partial}{\partial y} \\
0 & \frac{\partial}{\partial y} & \frac{\partial}{\partial x}
\end{bmatrix}^T.$$  

(3)

For linear problems the relation between stresses and strains is described by

$$\sigma = D(\varepsilon - \varepsilon_0),$$

(4)

with the prestrain vector $\varepsilon_0$. The linear elasticity matrix $D$ depends on the assumption of plane stress or plane strain conditions.

As usual, a function $u_{ex}$ with finite strain energy, fulfilling the geometric boundary conditions will be called weak solution of (1)–(4), if

$$B(u, v) = \mathcal{F}(v)$$

(5)
for all admissible functions \( v \). The symmetric bilinear form

\[
\mathcal{B}(u, v) := \int_{\Omega} (L u)^T D (L v) t_e \, dx \, dy
\]

is called the energy product of the plane elasticity problem. The strain energy is defined as

\[
\|u\|_{E(\Omega)}^2 = \mathcal{U}(u) := \frac{1}{2} \mathcal{B}(u, u)
\]

and the energy norm as a measure of \( u \) is

\[
\|u\|_{E(\Omega)} := \sqrt{\mathcal{U}(u)}.
\]

The linear load functional \( \mathcal{F}(v) \) can be decomposed into parts for volume loads, loads from prestrains and boundary loads

\[
\mathcal{F}(v) := \int_{\Omega} \left( X v_x + Y v_y \right) t_e \, dx \, dy + \int_{\Omega} (L v)^T D \varepsilon_{0} t_e \, dx \, dy + \int_{\Gamma_T} \left( \mathcal{T}_n v_n + \mathcal{T}_t n \right) t_e \, ds.
\]

A finite-element approximation to the exact solution is now constructed as

\[
u(x, y) = \sum_{i=1}^{m} N_i^j(x, y) a^j
\]

with shape functions \( N_i^j \) \((i = 1, \ldots, m)\) being defined on triangular or quadrilateral elements of diameter \( h \) and maximum polynomial order \( p \). As usual, each (vector valued) shape function \( N_i^j \), corresponding either to a displacement mode in \( x \)- or \( y \)-direction, is formed from a scalar shape function \( \tilde{N}_k \) by \( N_i^{2k-1} = (N_i^k, 0)^T \) and \( N_i^{2k} = (0, N_i^k)^T \).

The principle of virtual work (5) can be formulated for the approximation (10) and a finite number of test functions \( N_i^j \):

Find coefficients \( a^j \) \((i = 1, \ldots, m)\) so that geometric boundary conditions are fulfilled and that

\[
\mathcal{B} \left( \sum_{i=1}^{m} a^i N_i^j, N_i^j \right) = \mathcal{F}(N_i^j),
\]

holds for all test functions \( N_i^j \) \((i = 1, \ldots, m)\). This leads to a linear equation system for the unknown coefficients \( a^j \) with a symmetric, positive definite stiffness matrix

\[
K := (k^{ij})_{(i,j=1,\ldots,m)} := \left( \int_{\Omega} (L N_i^j)^T D (L N_j^j) t_e \, dx \, dy \right)_{(i,j=1,\ldots,m)}
\]

and the load vector

\[
f = \begin{pmatrix}
\left( \int_{\Omega} (N_i^j)^T (X t_e t_x) \, dx \, dy \right) \\
\left( \int_{\Omega} (N_i^j)^T (Y t_e t_y) \, dx \, dy \right) \\
\left( \int_{\Omega} (L N_i^j)^T D \varepsilon_{0} t_e \, dx \, dy \right) \\
\left( \int_{\Gamma_T} \mathcal{T}_n (N_i^j) t_e \, dx \right) \\
\left( \int_{\Gamma_T} \mathcal{T}_t (N_i^j) t_e \, dy \right)
\end{pmatrix}_{(j=1,\ldots,m)}
\]
With \( I_e = \{e_1, \ldots, e_n\} \) being the index subset of the shape functions of element \( e \) the formulation of the element stiffness matrix

\[
K_e := (k^l_{ej})_{l,j \in I_e} := \int_{\Omega_e} \left( L N_e \right)^T D \left( L N_e \right) t_e \, dx \, dy
\]

(14)

involves forming the matrix of element shape functions

\[
N_e := \left( N^l \right)_{l \in I_e} = \begin{pmatrix}
N^{k_1} & 0 & N^{k_2} & 0 & \ldots & 0 & N^{k_{n-2}} & 0 \\
0 & N^{k_1} & 0 & N^{k_2} & 0 & \ldots & 0 & N^{k_{n-2}}
\end{pmatrix}
\]

(15)

with \( n \) being the number of shape functions of the element. In \( p \)-version implementations, usually hierarchical shape functions are used, meaning that the shape function matrix \( N_{(e),p} \), corresponding to a polynomial degree \( p \) is a submatrix of the shape function matrix \( N_{(e),p+1} \).

We can define the \( B \)-matrix as the application of the differential operator \( L \) to the shape function matrix \( N_e \), i.e. \( B_e := L N_e \). The finite-element approximation of the displacement on element \( e \) is given by \( \bar{u}_e := N_e a_e \), \( a_e \) being the vector of solution coefficients corresponding to shape functions in element \( e \). With the relations between displacements, strains, and stresses, the strain of the finite-element solution reads \( \bar{\varepsilon}_e := \bar{L} \bar{u}_e = L N_e a_e = B_e a_e \) and the finite-element stresses can be computed by \( \bar{\sigma}_e := D \bar{\varepsilon}_e = D B_e a_e = D B_e a_0 \). Details for an implementation of the \( p \)-version finite-element method can be found in [28].

3. The overlay approximation

3.1. The basic idea in one dimension

Consider a one-dimensional model problem being approximated on a base mesh of three elements by high order polynomials as shown in Fig. 1. Let us now assume the exact solution of the problem to be smooth in an area covered by the first and last element, but to have a strongly varying character in the central element. Although the \( p \)-version will be able to represent the smooth part of the solution accurately, it will not be able to yield sufficient accuracy on the mesh chosen in the central part of the domain, where a refined mesh would be necessary. Instead of refining the \( p \)-mesh, we will now construct an independent second mesh and define an \( h \)-approximation with linear shape functions (Fig. 1) being superimposed on the basic \( p \)-version approximation (Fig. 2). This hierarchical sum of \( h \)- and \( p \)-shape functions will be \( C^0 \)-continuous on the entire domain if we only force the \( h \)-part of the solution to be zero at the boundary of the local mesh. It can be expected, that this hierarchical sum of ansatz spaces is now better suited to accurately represent the (global) smooth part and the (local) unsmooth part of the exact solution.

3.2. Formulation of the \( h p \)-d method in two dimensions

We now consider what on a first glance seems to be a geometrically incompatible finite element mesh on a domain \( \Omega \) shown in Fig. 3. There are two meshes, a \( p \)-version mesh on \( \Omega_1 \) and an \( h \)-version mesh on \( \Omega_2 \). Note that the \( p \)-version mesh is also defined under most of the \( h \)-version elements. The finite-element discretization consists of three parts, a region \( \Omega_1 \setminus \Omega_2 \) with \( p \)-version elements only, the region \( \Omega_1 \cap \Omega_2 \) of the hierarchical overlay of the \( p \)-elements with an \( h \)-version mesh and the region \( \Omega_2 \setminus \Omega_1 \) where the \( p \)-version mesh has a "hole", i.e. the structural behavior is represented by the \( h \)-version discretization only.
generalization to an earlier publication [22], where the hp-d method was formulated for a full overlay of meshes we will refer to discretizations where overlay elements do not always cover elements of the base mesh as hp-d method with partial overlay. Being precise, there exist triangulations

\[ T_1 = \{ t_{1,i}, i \in I_1 \} \quad \text{and} \quad T_2 = \{ t_{2,j}, i \in I_2 \} \]  

of the domains \( \Omega_1 \) and \( \Omega_2 \) respectively. A finite element space \( S_{p,\Omega_1} \) over \( T_1 \) is defined as the set of continuous functions being piecewise polynomials of order \( p \) over each element, vanishing on \( \partial \Omega_1 \cap \Gamma_{\nu}' \). In addition, \( S_{h,\Omega_2} \) shall denote a "standard" finite-element approximation using for example bilinear shape functions on \( T_2 \), vanishing on \( \partial \Omega_2 \cap \Gamma_{\nu}' \).

Let \( T_{21} = \{ t_{2,j}, j \in I_{2,1} \subset I_2 \} \) be the subset of elements of \( T_2 \) that are a triangulation of the domain \( \Omega_1 \cap \Omega_2 \). Then we assume that for each element \( t_{2,j} \), there exists a \( t_{1,i} \) with \( t_{2,j} \subset t_{1,i} \). This condition guarantees, that a "fine" element does not overlap edges of the coarse mesh. Let

\[ S_{h,\Omega_2}^0 := \{ v \in S_{h,\Omega_2}, v = 0 \ \text{on} \ \Gamma_2 \} \]
and
\[
S^0_{p,\Omega_1} := \{ v \in S_{p,\Omega_1} | v = 0 \text{ on } \Gamma_1 \}.
\] (18)

The finite element space of the hierarchical overlay is constructed by
\[
S_{h,p} = S^0_{p,\Omega_1} \oplus S^0_{h,\Omega_2}.
\] (19)

Every element \( v = v_p + v_h \in S_{h,p} \) is defined as the hierarchical sum of an \( h \)- and a \( p \)-function being \textit{continuous by construction}. This important observation is due to the homogeneous boundary values prescribed to the overlay mesh shape functions at the transition of the overlay to the \( p \)-version (\( \Gamma_2 \)) and respectively to the base mesh shape functions at the transition to the \( h \)-version discretization (\( \Gamma_1 \)). Now let \( \{ N^i_p | i \in I_p \} \) be the set of shape functions of the \( h \)-approximation on the fine mesh being collected in the shape function matrix \( N_h \) and \( \{ N^i_p | i \in I_p \} \) be the set of shape functions of the \( p \)-approximation on the coarse mesh composed to the shape function matrix \( N_p \). All shape functions are ordered so that the \( h \)-version shape functions get the highest numbers, i.e. the global shape function matrix gets the following structure:
\[
N := (N_p, N_h),
\] (20)

where \( N_h \) is appended to the matrix \( N_p \). The system matrix in the finite element formulation corresponding to Eq. (12) is defined as
\[
K := (k^i,j)_{i,j \in I_h \cup I_p} := \left( \int_{\Omega} (LN^i_p)^T D (LN^j_p) t_{ij} \, dx \, dy \right)_{i,j \in I_h \cup I_p}.
\] (21)

Using the hierarchical nature of \( S_{h,p} \), Eq. (11) is equivalent to the solution of the following problem:
\[
\text{Find } \mathbf{u} = u_h + u_p \in S_{h,p} \text{ so that for every } v_h \in S^0_{h,\Omega_2}, v_p \in S^0_{p,\Omega_1}
\]
\[
\mathcal{B}(u_p + u_h, v_p) = \mathcal{F}(v_p), \quad \mathcal{B}(u_p + u_h, v_h) = \mathcal{F}(v_h).
\] (22)
The resulting linear equation system also reflects the hierarchical nature of this formulation:

\[
\begin{bmatrix}
K_{pp} & K_{ph} \\
K_{hp}^T & K_{hh}
\end{bmatrix}
\begin{bmatrix}
a_p \\
a_h
\end{bmatrix} =
\begin{bmatrix}
f_p \\
f_h
\end{bmatrix},
\]

(23)

where the indices \( p \) and \( h \) refer to the \( p- \) and \( h- \)version shape functions respectively.

Care has to be taken that no linear dependencies in the shape function spaces are present in the discretization. This can be easily achieved if the overlay is restricted to linear elements and all linear modes of base elements being covered by overlay elements are eliminated.

A direct implementation of this \( hp-d \) method involving modification of a standard \( p \)-code and its data structures was discussed in [21]. As an alternative to this direct implementation a block-iterative solution of (23) can be considered. It is a first important observation that, in the one-dimensional case, if linear shape functions on both base and overlay mesh are used, the equation system is completely decoupled (see [15]). In the two-dimensional case, if linear elements are used for base and overlay meshes again, this formulation corresponds to the well known hierarchical multilevel splitting developed by Yserentant [30] and is closely related to local defect correction methods described by [11]. There are also similarities to coarse/fine mesh preconditioners proposed in [4] and to domain decomposition preconditioners introduced in [18]. A strategy to solve this equation system without explicit computation of the coupling matrix \( K_{ph} \), using elements with arbitrary polynomial degree on the base mesh, will be described in the following section.

4. Implementation of the \( hp-d \) method with a block Gauss–Seidel iteration

4.1. The basic procedure

The solution of the coupled equation system (23) can be obtained efficiently in a block Gauss–Seidel iteration. The coupling matrices \( K_{pp} \) and \( K_{ph}^T \) are brought to the right-hand side of the equation systems and are considered together with the corresponding iterates as additional load terms

\[
K_{pp} a_p^{(i+1)} = f_p - K_{ph} a_h^{(i)}, \quad K_{ph} a_h^{(i+1)} = f_h - K_{hp}^T a_p^{(i+1)}. \tag{24}
\]

The terms \( K_{ph} a_h^{(i)} \) and \( K_{hp}^T a_p^{(i+1)} \) can be interpreted as load functionals from 'prestrains' resulting from the iterated displacement fields \( a_h^{(i)} \) and \( a_p^{(i+1)} \) respectively. To prove this, define

\[
\bar{\varepsilon}_0 = L N_h a_h \tag{25}
\]

to be a strain corresponding to a given displacement field \( a_h \), \( N_h \) being the matrix of all \( h \) shape functions and \( L \) denoting the linear strain operator. The discrete load vector resulting from \( \bar{\varepsilon}_0 \) is given by

\[
f_h = \int_\Omega (L N_h)^T D \bar{\varepsilon}_0 t_z \, d\Omega, \tag{26}
\]

\( D \) being the elasticity matrix. Inserting (25) in (26) and using the \( B \)-matrix in the standard notation as the strain operator \( L \) applied to the matrix of the shape-functions

\[
f_h = \int_\Omega B_p^T D B_h t_z \, d\Omega a_h \tag{27}
\]

we can now identify the product \( K_{ph} a_h \):

\[
K_{ph} a_h = \int_\Omega B_p^T D B_h t_z \, d\Omega a_h. \tag{28}
\]
The importance of this observation is that it is not necessary to compute the coupling matrices explicitly. Only the product with the corresponding displacement vectors is involved in iteration (24). This enables the use of different programs for $p$- and $h$-version computations. Furthermore, the hierarchical local–global finite-element approximation can be implemented around existing finite-element codes interpreting $\bar{\varepsilon}_{int}$ as a “prestrain-like” additional load term. While there are some minor changes necessary in the $p$-version code to correctly compute the load vector from the prestrains, the $h$-version component does not need to be changed at all. The $h$-version can be represented by a standard finite-element code capable of incorporating prestrains as variable loads. The only additional program module required is a postprocessor that is able to interpolate strains from one mesh to the other.

4.2. Computation of displacements, strains, stresses and strain energy

Displacements are computed as the product of the shape function matrix with the solution coefficients of the hierarchical equation system:

$$\bar{u} = Na = [N_p, N_h] \begin{bmatrix} \bar{a}_p \\ \bar{a}_h \end{bmatrix} = N_p\bar{a}_p + N_h\bar{a}_h = \bar{u}_p + \bar{u}_h.$$  \hfill (29)

Strains are obtained by applying the operator $L$ to the displacements

$$\bar{\varepsilon} = L\bar{u} = L(N_p\bar{a}_p + N_h\bar{a}_h) = L(N_p\bar{a}_p) + L(N_h\bar{a}_h) = \bar{\varepsilon}_p + \bar{\varepsilon}_h,$$ \hfill (30)

i.e. again by a superposition of the strains from the $p$- and $h$-version solutions. Stresses are computed from strains in the usual manner

$$\bar{\sigma} = D(\bar{\varepsilon} - \bar{\varepsilon}_0) = \bar{\sigma}_p + \bar{\sigma}_h$$ \hfill (31)

with $D$ being the elasticity matrix. Element strains and stresses in those $p$-elements having a hierarchical overlay will be discontinuous due to the jumps on the inter-element boundaries of the $h$-elements. It was already mentioned that it is possible to implement the $hp$-$d$ iteration using an existing $h$-version finite-element code that is able to incorporate prestrains as distributed loads. Yet one has to be careful when evaluating stresses. In a standard code, stresses are computed from the finite-element strains as

$$\bar{\sigma}_h = D(\bar{\varepsilon}_h - \bar{\varepsilon}_0),$$ \hfill (32)

where $\bar{\varepsilon}_0$ are given prestrains. Yet, during the $hp$-$d$ iteration, the term $\bar{\varepsilon}_h$ is only a pseudo-strain resulting from the coupling of base and overlay mesh and cannot be used to compute stresses from (32). The correct stresses of the $hp$-$d$ solution must be obtained in a postprocessing procedure from the correctly superimposed strains by (31).

The strain energy and the corresponding energy norm must also be computed in a postprocessing step from the hierarchical overlay of the $p$- and the $h$-version solutions:

$$\|\bar{u}\|_{E(\Omega)}^2 = U(\bar{u}) = \frac{1}{2} \mathcal{B}(\bar{u}, \bar{u}) = \frac{1}{2} \int_{\Omega} \bar{\sigma}^T \bar{\varepsilon} \, d\Omega = \frac{1}{2} \int_{\Omega} (\bar{\sigma}_p + \bar{\sigma}_h)^T (\bar{\varepsilon}_p + \bar{\varepsilon}_h) \, d\Omega$$

$$= \mathcal{B}(\bar{u}_p, \bar{u}_p) + 2\mathcal{B}(\bar{u}_p, \bar{u}_h) + \mathcal{B}(\bar{u}_h, \bar{u}_h).$$ \hfill (33)

Because of the coupling terms $2\mathcal{B}(\bar{u}_p, \bar{u}_h)$ the strain energy terms of base and overlay solution can not simply be added from separately computed energies of the $p$- and the $h$-solution. This is because $\mathcal{B}$ is a bilinear, not a linear form.
4.3. Acceleration of the block iteration and adaptive mesh construction

The block Gauss–Seidel iteration (24) converges sufficiently fast if a good initial solution of the iterates \( u_h \) and \( u_p \) can be chosen. In the case of a full overlay of the \( h \)- and \( p \)-mesh, it is straightforward to choose \( u_h^{(0)} = 0 \) as the overlay solution is only a local correction to the base solution \( u_p \). Choice of an initial solution is more difficult in the case of a partial overlay because in this case \( u_h \) has to approximate \( u_{\text{ex}} \) in \( \Omega_1 \setminus \Omega_2 \) and has to correct \( u_p \). Therefore an initial iterate \( u_h^{(0)} = 0 \) often results in a large number of block iterations. It is yet possible to find an excellent initial solution for the iteration at almost no additional computational costs. The idea is to compute the initial solution by the \( hp \)-iteration itself on a very coarse discretization. The degree \( p \) on the base mesh is set to \( p = 1 \) or \( p = 2 \) and a very coarse overlay mesh is generated. It even can be efficient to replace the load \( f \) by an approximate \( \tilde{f} \) that is an integral equivalent in size and direction. The initial solution is then computed by an \( hp \)-iteration with the number of degrees of freedom for base and overlay systems being very small. Therefore almost no computational effort is needed, even if this iteration needs many steps to converge, yielding an initial estimate \( \tilde{u}_h \) and \( \tilde{u}_p \). These approximations \( \tilde{u}_h \) and \( \tilde{u}_p \) can be used to initialize a refined \( hp \)-iteration and to construct an adapted mesh for the overlay domain \( \Omega_2 \).

To get an a posteriori error estimation yielding error indicators for an adaptive refinement of the overlay mesh, we consider again the weak formulation of the \( hp \)-problem (22). As the major concern of this work is an accurate approximation of the local problem, we will focus on indicators for the local error, only. The error on the base mesh can e.g. be estimated by extrapolation of results from a \( p \)-extension.

There are two coupled systems for the base and the overlay mesh. The system

\[
B(u_p + u_h, v_h) = F(v_h) \quad \forall v_h \in S_0^{h, \Omega_2} \tag{34}
\]

is equivalent to the following, weakly formulated boundary value problem on the local domain \( \Omega_2 \):

Find \( u_h \) on \( \Omega_2 \) with \( u_h = 0 \) on \( \partial \Omega_2 \cap \bar{\Gamma}_u \cup \Gamma_2 \) so that

\[
B(u_1, v_h) = F(v_h) \quad \text{with} \quad F(v_h) = F(v_h) - B(u_p, v_h), \tag{35}
\]

for all \( v_h \) of the test space \( S_0^{h, \Omega_2} \).

Because Eq. (35) is a low order, standard finite-element approximation of (34), the discretization error of (34) can be estimated by well known a posteriori procedures (e.g. [1–3,13,14]), under the assumption that \( u_p \) or an approximation \( \tilde{u}_p \) be given, which can be obtained from an initial iteration of the coupled system (22). Our numerical experiments show, that even a coarse approximation \( \tilde{u}_p \) is sufficient for constructing error indicators that are able not only to reflect all local features of the solution but also to adequately control local mesh refinement (see the numerical examples in Section 6).

4.4. An algorithm for the iterative \( hp \)-method

The above procedure is summarized in Algorithm 1. In the initial step 1 a first \( hp \)-finite-element discretization is set up. Base and overlay regions are identified and a coarse overlay mesh is generated. In step 4 the stiffness matrices for the \( p \)-version system and the \( h \)-version system are computed independently. Homogeneous boundary conditions are prescribed on the boundary \( \Gamma_2 \) of the overlay mesh and \( \Gamma_1 \) of the base mesh. In the first iteration step the incremental load vector for the \( p \)-version on the base mesh is set to zero. In subsequent iterations the incremental right-hand side for the base system is computed in step 9 from the strains of the \( h \)-version solution and subtracted from the original load vector. After solving for the \( p \)-version system with the actual load vector in step 11, an incremental right-hand side for the overlay system is computed from the strains of the \( p \)-version solution and subtracted from the original load vector \( f_h^{(i)} \) in step 12. The \( i \)th iterated solution of the \( h \)-version system is obtained in step 13. LU-decomposition of
the system matrices is possible, so that during the iteration in steps 11 and 13 only backsubstitutions for the modified load vectors have to be performed. If the iteration is performed to compute the final solution, error estimation is executed on the overlay in step 16. If the estimated error is larger than a user defined threshold, the overlay mesh is refined according to the error indicator (step 18), the stiffness matrix has to be recomputed (step 19) and the overlay equation system has to be solved again (step 20). Steps 16 to 20 are repeated until the specified accuracy is reached. Next in step 24 the overall solution is obtained by hierarchical overlay of \( \hat{\mathbf{u}}_p^{(i)} \) and \( \hat{\mathbf{u}}_h^{(i)} \), and stresses, strains and strain energy are computed. In case of convergence the iteration is stopped.

Algorithm 1. Adaptive hp-d algorithm.
1: create coarse overlay mesh, set \( p = 1 \) on base mesh
2: set compute_start_solution true
3: set \( i = 0 \)
4: compute \( K_{pp} \) and \( K_{hh} \)
5: repeat
6: if \( (i = 0) \) then
7: \( \hat{\mathbf{f}}_p^{(i-1)} = 0 \)
8: else
9: compute load increment \( \hat{\mathbf{f}}_p^{(i-1)} = \int_\Omega (LN_p)^T \mathbf{D}_h^{(i-1)} t_z \mathrm{d}\Omega \)
10: end if
11: solve \( K_{pp} \hat{\mathbf{u}}_p^{(i)} = \hat{\mathbf{f}}_p - \hat{\mathbf{f}}_p^{(i-1)} \)
12: compute load increment \( \hat{\mathbf{f}}_h^{(i)} = \int_\Omega (LN)_h^T \mathbf{D}_p^{(i)} t_z \mathrm{d}\Omega \)
13: solve \( K_{hh} \hat{\mathbf{u}}_h^{(i)} = \hat{\mathbf{f}}_h - \hat{\mathbf{f}}_h^{(i)} \)
14: if (not compute_start_solution) then
15: repeat
16: compute error indicators on \( \Omega_2 \)
17: if (estimated error on \( \Omega_2 \) is too large) then
18: refine the mesh according to error indicators
19: compute \( K_{hh} \)
20: solve \( K_{hh} \hat{\mathbf{u}}_h^{(i)} = \hat{\mathbf{f}}_h - \hat{\mathbf{f}}_h^{(i)} \)
21: end if
22: until (estimated error on \( \Omega_2 \) is smaller than threshold)
23: end if
24: compute displacements, strains, stresses and strain energy
25: check convergence
26: \( i = i + 1 \)
27: until (converged)
28: if (compute_start_solution) then
29: compute error estimator for base mesh and increase \( p \)
30: create detailed overlay mesh
31: set compute_start_solution false
32: goto step (4):
33: end if

If the iteration computed only an initial solution, the discretization is now refined. In step 29 the polynomial degree of the shape functions on the base mesh is increased and an overlay mesh is created.
accounting for all necessary geometrical and physical details. The iteration is restarted with step 4 but now the information of the initial solution is contained in $\mathbf{u}_h^{(i-1)}$.

5. Benchmark problems

5.1. L-shaped domain

To verify the correctness of the $hp$-d iteration a benchmark problem with known exact solution is investigated. As a typical two dimensional problem with singularities, the L-shaped domain shown in Fig. 4 is considered. The convergence of the iterative solution procedure can be studied and the accuracy of various $hp$-d discretizations can be investigated.

Boundary conditions are prescribed so that the analytic solution can be computed and corresponds to a MODE 1 loading of the structure. At the reentrant corner the exact solution is characterized by a term

$$u_1(r, \Theta) = A_1 r^\lambda_1 \psi_1(\Theta).$$

Here polar coordinates $r$ and $\Theta$ are centered at the reentrant corner and the function $\psi_1(\Theta)$ is analytic so that the smoothness of the solution is characterized by the exponent $\lambda_1$. The factor $A_1$ is closely related to a stress intensity factor [27,29]. The exact solution of the problem and the derivation of the boundary conditions are discussed in detail in [28, p. 173 ff], where also $\lambda_1 = 0.544483737$ is obtained. Plane strain conditions are assumed, Poisson's ratio is 0.3 and Young's modulus $E$ and the thickness $t$ are set to 1.0. The strain energy of the exact solution is computed to $\|u\|_E^2 = 4.15454423$.

Fig. 4. L-shaped domain (from [28]).
5.1.1. hp-d discretizations

The base mesh, constructed with one layer of refinement at the reentrant corner, having a progression factor of 0.15, is shown in Fig. 5(a). To study the convergence of the iterative procedure, four different overlay meshes are generated, all covering the elements adjacent to the singularity with numbers 1, 2 and 3 in the base mesh. Three uniform meshes are derived by subdividing every base mesh element into $4 \times 4$, $8 \times 8$ and $16 \times 16$ overlay elements. These meshes have 48, 192 and 768 elements and the characteristic element size of the overlay mesh is $h = 0.0375$, $h = 0.01875$ and $h = 0.009375$, respectively. They will be further referred to as (h4), (h8) and (h16). The fourth overlay mesh was generated with a density function controlled mesh generator [23], where the element size $h$ could be prescribed at the nodes of an arbitrary background mesh. In this case the $8 \times 8$ mesh (h8) was used for defining the mesh density and the element size was set to $h = 0.002$ at the singularity, increasing to $h = 0.035$ in the rest of the domain. The mesh generated according to this density function with 444 elements will be further referred to as (ha). The complete discretizations with base and overlay meshes (h4) and (ha) are shown in Fig. 5(b) and (c). In addition, a zoomed plot of the overlay mesh (ha) is shown in Fig. 5(d).

Fig. 5. hp-d discretization L-shaped domain: (a) base mesh, (b) base and $4 \times 4$ overlay mesh (h4), (c) base and refined overlay mesh, (d) refined overlay mesh 444 elements.
5.1.2. hp-d boundary conditions
The L-shaped domain is loaded with the tractions from the MODE 1 stress tensor. Only boundaries of the base mesh have to be loaded, as the overlay elements are located at the stress free edges. From the theoretical background of the method as a special construction of a conforming ansatz space we can now formulate the following modeling rules for the boundary condition:

1. Both meshes have to fulfill the prescribed boundary conditions of the overall mechanical system. If an edge of the system is clamped, boundaries must be clamped on both the base and the overlay model. The same holds for free edges and edges with symmetry conditions.
2. On the interface boundary \( \Gamma_2 \), i.e. at transitions from the hp-d discretization to the p-version, homogeneous boundary conditions have to be prescribed for the h-version approximations to guarantee \( C^0 \) continuity of the overall solution. In general, this interface is a set of nodal points in a one-dimensional, of edges in a two-dimensional and of faces in a three-dimensional case.
3. Special care has to be taken considering nodal modes of the base mesh being associated to nodes in the interior of \( \Omega_2 \). These modes can either be expressed directly by a p ansatz function or by a linear combination of h ansatz functions on the overlay. To eliminate linear dependencies of the global system, the corresponding degrees of freedom of \( \mathbf{u}_p \) have to be set to zero.

According to these rules all boundary conditions for the example are prescribed. Along the stress free inner edges natural boundary conditions for the base mesh and for the overlay mesh are set. This allows the overlay solution to also add contributions to the ansatz space. To eliminate rigid body modes, the displacement of the node at the reentrant corner is set to zero.

5.1.3. Convergence of the block iteration
In Figs. 6–9 the relative iteration error of the hp-d-solution \( \eta_{rel} = (\|\mathbf{u}_{ref}\|^2 - \|\mathbf{u}_l\|^2) / \|\mathbf{u}_{ref}\|^2 \) is plotted against the number of iteration steps for various \( p \) on the base mesh. Because the iterative implementation does not provide means for direct solution of the equation system, the strain energy of the solution after

![Fig. 6. Convergence of the hp-d iteration for uniform mesh h4.](image-url)
50 iteration steps, where the iterated solution is stable up to six digits, is taken as the reference value $\|\bar{u}_{\text{ref}}\|^2$. The most important result is that the iteration converges very fast for all tested combinations of $p$ and $h$ meshes during the first iteration steps. In all cases the relative iteration error is less than 1 percent after only three steps. The observed fast convergence for this example is due to the good initial solution $\bar{u}_b^{(0)} = 0$ for the overlay mesh. For the refined overlay mesh (ha) and a base mesh with $p = 2$, the starting value of the strain-energy of the $p$-version solution was $|\|u_p^{(1)}\|^2_E = 4.0735$. The energy of the $hp$-$d$ solution after the first
iteration was $\|\varepsilon_h^{(3)}\|_E^2 = 4.0244$. After three iteration steps a value of $\|\varepsilon_h^{(3)}\|_E^2 = 4.1266$ corresponding to an iteration error of 0.14% was obtained. The relative iteration error after 10 iteration steps was 0.0030426%.

5.1.4. Convergence of the discretization

We now investigate the convergence rate in energy norm for the hp-d-discretization and compare it to the well known h-, p-, and hp-extensions of the finite-element method. In Fig. 10 the effect of increasing $p$ on the base mesh can be studied. All curves represent a different overlay mesh. As a reference, the typical S-shaped curve of a p-extension on a geometrically refined mesh and the convergence of an adaptive $h$-refinement using bilinear elements are also shown. Curves of the discretizations with finer meshes (h8), (h16) and (ha) are descending very fast for low $p$, as at the beginning of the extension process the dominant part of the error is still in the smooth part of the solution. Therefore the high convergence rate of the $p$-version for
smooth solutions is observed, without being perturbed by the singularity. Yet, as soon as the error at the singularity becomes dominant, the convergence rate slows down to the algebraic rate $\lambda_1$ of the $p$-version. Discretization with mesh (h4) leaves a stronger influence from the singularity already from the beginning of the extension process and the curve is therefore more flat, still slowing down to the asymptotic convergence rate of the $p$-version. The combination of the locally refined overlay mesh (ha) and a base mesh at $p = 3$ and 1019 degrees of freedom yield a relative error in energy norm of 3.1%.

A fixed polynomial degree $p = 3$ on the base mesh is now used for investigation of $h$-extensions on the overlay. The three curves in Fig. 11 represent a uniform $h$-extension on the overlay with meshes (h4), (h8) and (h16), a graded $h$-extension on the overlay with meshes (h8) and (ha) and, as a comparison of the efficiency, a third curve resulting from a standard uniform $h$-extension on the whole domain at $p = 3$.

The curves for the standard and $hp$-$d$ uniform $h$-extensions have the same slope $\beta = \lambda_1/2$ and the relative error in energy norm for the uniform $hp$-$d$ $h$-extensions is less than half of that for the standard $h$-extensions, because the singularity is better resolved by $hp$-$d$ discretizations. The curve for the graded $hp$-$d$ $h$-extension has a steeper slope. In the optimal case, i.e. with adaptive mesh refinement on the overlay, a convergence rate of $\beta = p_h/2$ can be reached, $p_h$ being the polynomial degree of the approximation on the overlay.

Summarizing, the $hp$-$d$-extension with mesh adaptation on the overlay is superior to uniform or adaptive $h$-extensions and to a $p$-extension on a uniform mesh, but it cannot reach the efficiency of a $p$-extension on a geometric mesh with its exponential presymptotic range of convergence. Therefore, the type of problems, where $hp$-$d$-extensions should be considered are those, where the local geometry of a structure is too complex to be resolved by a small number of $p$-elements with a geometric refinement to singular points, and where this local geometry is imbedded in a structure on a much larger geometric scale. An example will be given in the following section.

5.2. Single foundation problem

Following e.g. [16], a domain decomposition using an overlapping Schwartz approximation is equivalent to a Block–Gauss–Seidel iteration similar to the scheme outlined in Section 4. A major difference of the approach presented in this contribution is the hierarchical nature of the overlay. Yet, the similarity of the approaches and results of [16] suggest that the convergence properties of the iteration (24) depend on the width of the overlap of $\Omega_1$ and $\Omega_2$. This will be investigated in the following numerical study. The
The benchmark problem used is a model of a single foundation on an elastic half space as shown in Fig. 12. Symmetry boundary conditions are prescribed along the y-axis, the lower and right boundaries are clamped, i.e. $\hat{u} = \hat{v} = 0$ and the upper boundary is free. A reference solution is computed using the $p$-version of the finite element method on a geometric mesh. For this purpose, the singularity at the end of the line load is resolved with two layers of mesh refinement. From the results of a $p$-extension with $p = 7, 8$ and $9$ the strain energy of a highly accurate solution can be computed by Richardson extrapolation to a value of $\|\mathbf{u}\|_{E(\Omega)}^2 = 603.182385$.

Fig. 12. Single foundation on elastic half space.

Fig. 13. Base mesh with seven elements.
5.2.1. \( hp-d \) discretization

We choose an \( hp-d \) discretization consisting of a base mesh with 7 elements and a uniform overlay mesh of \( 20 \times 20 \) elements as shown in Figs. 13 and 14 respectively. To study the convergence behavior of the iterative solution method the width of the overlap of \( h- \) and \( p- \) meshes is successively decreased. For each sample problem the size of base mesh elements 5, 6 and 7 is decreased by the size of two overlay elements. The limiting cases, with the overlay being 10 elements and 2 elements wide, are depicted in Fig. 14(a) and (b). The problem is solved for polynomial degrees of \( p = 3 \) and 5 on the base mesh.

![Graph](image-url)  
Fig. 15. Convergence of the \( hp-d \) iteration for \( p = 3 \) on the base mesh.
5.2.2. Convergence of the block iteration

The results of this parameter study in terms of the relative iteration error of the \(hp-d\)-solution, 
\[ \eta_{rel} = \frac{(\|\tilde{u}_{\text{ref}}\|^2 - \|\tilde{u}\|^2)}{\|\tilde{u}_{\text{ref}}\|^2}, \]
plotted over the number of iteration steps can be seen in Fig. 15 for \(p = 3\) on the base mesh and in Fig. 16 for \(p = 5\) respectively. As in the previous example, a reference value was taken when the strain energy of the solution was stable up to five digits. Each curve describes the convergence of the iterative solution method for a specific width of the overlap. It can be clearly seen that the rate of convergence of the block iteration becomes smaller when the width of the overlap is decreased. For the case of a partial overlap with a small width of overlap, this result strongly supports the necessity of an acceleration of the iteration by an improved initial solution, as outlined in Section 4.3.

As a further result we observe that, comparing Figs. 15 and 16, no influence on the convergence rate of the degree of the polynomial shape functions used on the base mesh can be identified.

Finally, we remark that by varying the width of the overlap in the \(hp-d\) method, not only the domain decomposition has changed, as this would be the case in a classical additive Schwartz method, but that the approximation space itself also changes. It was therefore an expected result, that all iterations converge to a slightly different strain energy.

6. A complex example

We present an application of the \(hp-d\) method to a soil mechanics problem. The foundation of a building with a concrete parking garage shall be investigated. Due to a change in the properties of the underground an area of 300×200 m has to be accounted for in the computational model. In addition, interactions with a second, nearby located surface load shall be investigated in the same computational model. Special emphasis shall be laid on the modeling of interaction between soil deformation and the concrete foundation slab of the building.

The global structure with dimensions, material properties, loads and boundary conditions is shown in Fig. 17(a). A zoom of the building foundation is given in Fig. 17(b). To approximate the friction behavior
between the soil and the concrete wall a small layer of orthotropic material (region 2 in Fig. 17(b)) is introduced, where the stiffness in y-direction is half the value of that in x-direction.

6.1. hp-d discretization

An hp-d discretization with partial overlay depicted in Fig. 18 is introduced for this problem. The base p-version finite-element mesh shown in Fig. 18(a) consists of 11 elements only. The curved edges of the elements modeling the soil layer are quadratic parabolas, being mapped by the blending function method [28]. As shown in Fig. 18(b), the overlay, being described in more detail later, covers the upper left corner of the domain. According to the procedure outlined in Section 4 an initial solution is computed on a coarse discretization first. The polynomial degree of the elements of the base mesh (Fig. 18(a)) is chosen to be $p = 4$ and a coarse overlay mesh of 62 elements as shown in Fig. 19(a) is generated. Note again that, because this mesh is used only for the purpose of obtaining a good initial solution, it is not necessary to resolve all geometrical or material details. In this coarse model the load of the original system is substituted by an equivalent distributed load of $\bar{p} = 36.89$ [kN/m²] being applied to the edges of the two elements in the upper left corner of the mesh in Fig. 19(a).

After the first hp-d iteration has converged to a sufficient accuracy a new overlay mesh is generated, accounting now for all geometrical and physical details (Fig. 19(b)). A priori criteria lead to a refinement towards the various singular points in the computation domain. In the following iteration the polynomial degree on the base mesh is increased to $p = 8$. The results of the initial solution are transferred directly from the coarse to the refined overlay mesh, imposing pseudo-prestrains as computed in the initial solution phase to the refined overlay mesh.

After each finite-element computation on the overlay mesh, error estimation controls the accuracy of the overlay solution and the generation of an adaptively refined overlay mesh. Only two block iteration steps
on the refined overlay mesh were necessary to achieve an accuracy which we consider sufficient for engineering purposes. To demonstrate that no significant further changes in stresses and displacements are observed, three more iteration steps were performed on the final system, where the overlay mesh was not further refined in the last two steps.

6.2. Convergence of the block iteration

The convergence of the enhanced $hp$-$d$ iteration is shown for the strain energy $\|\mathbf{u}\|_F^2$ in Fig. 20 and for the maximum displacement $\max(\text{abs}(u_y))$ measured at the upper left corner $A$ of the domain of computation in
Fig. 20. Convergence of strain energy.

Fig. 21. Convergence of minimum displacement.

Fig. 21. In both figures the solid line shows the convergence of the refined system, while the dashed line represents the coarse system. Each block iteration step of the coarse system consists of the solution of two equation systems of 151 unknowns on the base and 118 unknowns on the overlay mesh respectively. An iteration step on the refined mesh needs the solution of two equation systems with 611 unknowns on the base mesh and up to 1817 unknowns on the overlay, showing that the computational cost of the initial coarse solution phase is negligible. The strain energy of the $hp$-$d$ solution after the fifth iteration step was $\|\tilde{\mathbf{u}}_5\|^2 = 4.13239$, with a relative deviation between steps 4 and 5 of 0.12%.
6.3. Displacements and stresses

As expected, the maximum displacement of the coarse system converges to a value \( \max(\text{abs}(u_x)) = 0.0161 \, [\text{m}] \) significantly differing from the maximum displacement of the refined system \( \max(\text{abs}(u_x)) = 0.0179 \, [\text{m}] \), accounting for additional deformations of the concrete building structure and for the detailed model of the foundation itself.

A contour-plot of displacement \( u_y \) on the deformed structure is shown in Fig. 22(a) for the global system and as a zoom of the local situation with the building and its foundation in Fig. 22(c). The base mesh and the geometry of the overlay region are shown as dashed lines for reference and better orientation. Details like the deformation of the foundation plate and the concrete wall can now be observed. As expected, the solution is \( C^0 \)-continuous at the transition from the overlay range to the pure \( p \)-discretization.

Fig. 22. Results on the global and local sizescale: (a) displacements \( u_x \), (b) stresses \( \sigma_{yy} \), (c) displacements \( u_y \), (d) stresses \( \sigma_{yy} \).
Stresses $\sigma_y$ are shown on the deformed structure in Fig. 22(b) and (d). On the global structure (Fig. 22(b)), where the iso-surface plot displays a range of stresses between $-15.0$ and $+0.0 \text{ [kN/m}^2\text{]},$ a pattern typical for foundation stresses can be observed. On the local structure (Fig. 22(d)) stresses are displayed in the range from $-50.0$ to $+50.0 \text{ [kN/m}^2\text{].}$ Details of the solution like stress concentration under the concrete building wall are now resolved by the $h$-approximation on the overlay mesh. Bending stresses in the building floor plate due to the deformation of the soil are also observable.

7. Conclusions

The following conclusions can be drawn from the results:

- The method presented in this contribution is a hierarchical formulation using $h$- and $p$-version shape functions and yielding a conformal finite-element method. The resulting linear equation system can be solved by a block Gauss–Seidel iteration.
- The iteration leads to an engineering interpretation of the method in the sense that two different parts of the solution are approximated by two different finite-element extensions. Both discretizations may be fitted optimally to the character of the solution using a priori information. Smooth and nonsmooth parts of the solution can each be discretized by independent, specially suited shape function spaces.
- Although information is exchanged between both discretizations in an iterative scheme, the two coupled problems can be dealt with in independent computer codes.
- The combination of a $p$-version on a fixed mesh and an overlayed $h$-version on a successively refined mesh does not show the exponential convergence rate of $hp$-extensions. Nevertheless the results could be of practical importance, as they offer a compromise between modeling effort and convergence rate, especially if structures with strongly differing inherent length scales are to be investigated.

References


