



**HAL**  
open science

# Strategies around the local defect correction multi-level refinement method for three-dimensional linear elastic problems

Laureline Barbié, Isabelle Ramière, Frédéric Lebon

► **To cite this version:**

Laureline Barbié, Isabelle Ramière, Frédéric Lebon. Strategies around the local defect correction multi-level refinement method for three-dimensional linear elastic problems. *Computers & Structures*, 2014, 130, pp.73-90. 10.1016/j.compstruc.2013.10.008 . hal-01007370

**HAL Id: hal-01007370**

**<https://hal.science/hal-01007370v1>**

Submitted on 30 May 2018

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Strategies involving the local defect correction multi-level refinement method for solving three-dimensional linear elastic problems

L. Barbié<sup>a,b</sup>, I. Ramière<sup>a,\*</sup>, F. Lebon<sup>b</sup>

<sup>a</sup>CEA, DEN, DEC, SESC, F-13108 Saint-Paul Lez Durance, France

<sup>b</sup>LMA, CNRS, UPR 7051, Aix-Marseille Univ, Centrale Marseille, 31, Chemin Joseph Aiguier, F-13402 Marseille Cedex 20, France

The aim of this study was to assess the efficiency of the local defect correction multi-grid method (Hackbusch, 1984 [31]) on solid mechanics test cases showing local singularities and derived from an industrial context. The levels of local refinement are automatically obtained recursively, using Zienkiewicz and Zhu's *a posteriori* error estimator. Choices of the prolongation operator, the refinement ratio and criterion are discussed in order to give the most satisfactory performances. Comparisons with an h-adaptive refinement method show the efficiency of the tool presented here, in terms of its accuracy and the memory space and processor time required.

## 1. Introduction

Industrial simulations involve increasingly complex physics, generally related to various characteristic length-scales. The following two main approaches have been developed to generate locally refined meshes with fewer degrees of freedom (DoF) instead of adopting a uniformly refined mesh matching the finest singularity:

- The methods of the first kind, which are known as “adaptive methods”, consist in locally enriching recursively an initial mesh of the domain. As a consequence, the initial mesh has not to be fitted to the singularities. There exist four main adaptive refinement methods: the r-adaptive technique (e.g. [1,2]), the h-adaptive technique (e.g. [3–7]), the p-adaptive technique (e.g. [8–10]) and the s-adaptive technique (e.g. [11,12]). The aim of the first three approaches is to reduce the discretisation error by making local changes in either the position of the nodes (in the case of the r-adaptive method) or the number of DoF (in that of the h-adaptive method) or the degree of the polynomial basis functions (in that of the p-adaptive method). The main advantage of these latter adaptive methods is that the problem is eventually solved on a single optimum mesh. These methods are the most performed these days, especially in combined

versions giving the advantages of two or more methods, such as the hr-adaptive process [13,14] or the hp-adaptive process presented in [15–17], which is widely used in the fields of thermo-hydraulics [18,19], combustion [20], neutronics [21,22] and solid mechanics [23,24], for example.

However, some extra work on the solver is usually required (non-conforming meshes, preconditioning, etc.), and the resulting number of DoF in the problem may still be prohibitive in industrial contexts.

The s-adaptive method, which is slightly different from the other approaches, consists in overlaying the initial mesh with additional finer local meshes. A composite problem is then defined, accounting for the behaviour of each mesh and the interface coupling between the connected levels. The resulting number of DoF quickly becomes huge, which explains why this method is less widespread than the others of its kind. However, a version called the Arlequin method (e.g. [25,26]), in which domains with different behaviours (for example, 1-D and 2-D models or continuum and atomic models) are combined, has recently become quite popular in the field of solid mechanics.

- The methods of the second kind, which are called local multi-grid (or multi-level) methods (e.g. [27–29]), can be seen as s-adaptive methods where the problems defined on each grid are solved separately. A multi-grid process [30] based on prolongation and restriction operators enables the solutions obtained at each level to be connected to each other. Unlike the standard multi-grid approach, the local multi-grid procedure starts with a coarse overall grid (which is generally

\* Corresponding author. Tel.: +33 442253038.

E-mail addresses: barbie@lma.cnrs-mrs.fr (L. Barbié), isabelle.ramiere@cea.fr (I. Ramière), lebon@lma.cnrs-mrs.fr (F. Lebon).

non-data-fitted) covering the whole domain and then works recursively on fine local nested sub-grids. Each mesh is generally chosen structured regular (even Cartesian if it is possible). Several local multi-grid methods have been developed, which differ in terms of the restriction operator used to correct the next coarser solution: the Local Defect Correction (LDC) method [31], the Flux Interface Correction (FIC) method [32] and the Fast Adaptive Composite (FAC) method [33].

The most obvious advantage of the local multi-grid approach over the standard multi-grid one is the gain in terms of CPU time and memory space when processing local singularity problems. However, as with the standard multi-grid approach, an iterative process has to be performed in order to obtain a converged solution at each level. The accuracy of the solution is therefore strongly dependent on the precision of the projection operators.

The performances of all these refinement methods depend strictly on the possibility of accurately detecting the zone of interest, i.e. the zone where the maximum discretisation error is present. To automate the refinement process, these methods are therefore often combined with *a posteriori* error estimators (e.g. [7,3,16,22,34,35]).

One category of error estimators, in which comparisons are made between two different grid refinements (e.g. [36,37,35]), are based on the principle that the discretisation error converges with the mesh step. These tools can be applied to physical situations of all kind.

Most of the *a posteriori* error estimators which are specifically dedicated to solid mechanics problems (e.g. [38–40]) are based on the fact that the classical finite element (FE) solution does not locally satisfy the static admissibility equation. The greater part of these estimators has been developed and proved in the context of linear behaviours. However, they can usually be extended to nonlinear behaviours (e.g. [34,41]), but are often unsuitable for dealing with the most complex situations (plasticity, large deformations, contact, friction, etc.). Further details about these estimators can be found in [42].

In this paper, the test cases studied were based on industrial problems including local singularities with different characteristic length-scales (see Section 2.1). The main constraint of this work was to develop a refinement strategy that can be easily implemented in any existing industrial software. In this context, we decided to perform a method relevant in a “black-box” solver context. Neither the local h-refinement method nor the p-refinement or the s-refinement strategies meet this criterion. In addition, we decided to use local multi-grid methods so that the memory space required for each resolution would not constitute an obstacle. This also enabled us to benefit from the excellent solver performances obtained on structured regular meshes. Among the existing local multi-level methods, the LDC method [31] was selected for this study because it seemed to be the most suitable method for dealing with solid mechanics problems involving local singularities.

Although the local multi-grid concept has been widely applied in the fields of thermodynamics and thermal hydraulics (e.g. [43–46]), only a few recent studies have focused on this approach in the context of solid mechanics [35,47]. The method presented here is fairly similar to that described in [35] but we proposed to use an existing convergence proved local multi-grid method [27,48,49], whereas Biotteau et al. adapted the Full Multi-Grid method [50] to deal with local refinement problems. The iterative processes used therefore differ in terms of the prolongation operators as well as the resolution steps.

This paper is organised as follows. In Section 2, the industrial test case studied and the problems involved are presented. The local defect correction (LDC) multi-level method is then presented in

Section 3. Section 4 describes some strategies used to obtain accurate results with the LDC method. The accuracy, mesh convergence and automatic refinement procedures are focused on by performing 2-D and 3-D simulations. Lastly, Section 5 focuses on the performances of the LDC method with a view to applying it in industrial contexts, especially by making some comparisons with the existing global h-refinement method.

## 2. Context of the study

### 2.1. The industrial test case

The pellet-cladding interaction (PCI) [51] occurs during irradiation in pressurised water reactors, which constitute the majority of French nuclear reactors. The fuel consists in these reactors of cylindrical uranium dioxide ( $UO_2$ ) pellets 8.2 mm in diameter and 13.5 mm high, which are piled up in a zircaloy cladding. During irradiation, the following two processes lead to the occurrence of PCI:

- The fuel pellets quickly crack during the first power increase (see Fig. 3, left). In addition, the fuel pellets swell and the cladding creeps due to the external pressure, which results in the occurrence of contacts between the pellets and the cladding. The pellet cracking process therefore results in discontinuous contacts.
- The deforming effects of the high temperature gradient on the pellets give them an hourglass shape (cf. Fig. 1). Contacts between the fuel and the pellets therefore occur first in front of the inter-pellet plane. High stress concentrations then develop all around the inter-pellet plane.

The localised stress concentrations combined with the discontinuous contacts with the pellets can lead to the failure of the cladding. It is therefore of great importance to be able to model PCI accurately because the integrity of the cladding, which is the first confinement barrier for the irradiated fuel, is at stake. Considerable research and development efforts are being made on this topic worldwide in order to understand the mechanisms possibly leading to PCI failure, as well as to design PCI resistant rods. It is difficult to perform complete 3-D simulations because too many DoF would be required to be able to model the local processes accurately. The strategy often used at present consists in using a structured conforming mesh locally refined around the PCI zone. The

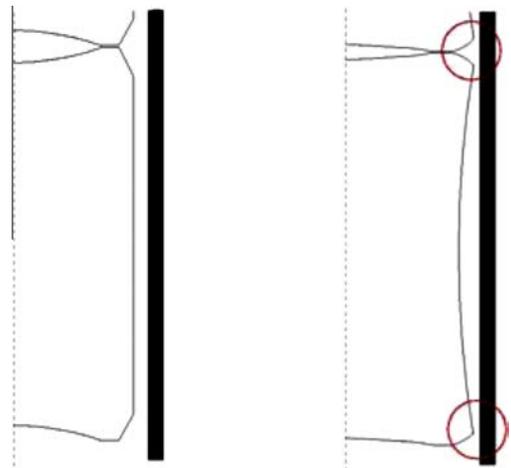


Fig. 1. Illustration of the hourglass shape deformation process: before (left) and during (right) irradiation.

local refinement includes locally highly stretched elements, which results in ill-conditioned systems, with which it is difficult to reach convergence. As described in the Introduction, the LDC method seems to make it possible to circumvent these difficulties and obtain more accurate solutions in a reasonably short CPU time.

In order to test the performances of the LDC method, a simplified PCI model is used in this study. The geometry is the same as the classical PCI simulation one. Pellet and cladding mechanical behaviours are simplified. We focus here on the response of the cladding, which is assumed to be linear elastic (Young's modulus:  $10^{11}$  Pa and Poisson's ratio: 0.3) (see problem (1)). We are therefore dealing with a linear problem in displacement. The contact with the pellet is simulated in the form of a discontinuous pressure imposed on the internal radius of the cladding.

The two characteristic mechanisms responsible for PCI are first modelled separately in two distinct two-dimensional studies (Sections 4.2 and 4.3). They are then approached together in a three-dimensional study (Section 4.4).

## 2.2. Mechanical formulation of the problem

Let us take a linear elastic domain  $\Omega$  with the boundary  $\Gamma$  and the outgoing normal  $\mathbf{n}$ . Prescribed displacements  $\mathbf{u}_0$  and forces  $\mathbf{F}$  are imposed on a subset  $\Gamma_U$  of the boundary and on the remaining part  $\Gamma_F$ , respectively, where  $\Gamma_U \cup \Gamma_F = \Gamma$  and  $\Gamma_U \cap \Gamma_F = \emptyset$ .

The problem to be solved can therefore be written as follows:

$$(\mathcal{P}) : \begin{cases} -\text{div}(\sigma) = \mathbf{f} & \text{in } \Omega & \text{(a)} \\ \sigma = \mathbf{C} : \varepsilon(\mathbf{u}) & \text{in } \Omega & \text{(b)} \\ \varepsilon(\mathbf{u}) = \frac{1}{2}(\text{grad}(\mathbf{u}) + \text{grad}^T(\mathbf{u})) & & \text{(c)} \\ \mathbf{u} = \mathbf{u}_0 & \text{on } \Gamma_U & \text{(d)} \\ \sigma \cdot \mathbf{n} = \mathbf{F} & \text{on } \Gamma_F & \text{(e)} \end{cases} \quad (1)$$

where

$$\begin{cases} \sigma & \text{is the stress field} \\ \mathbf{f} & \text{is the source term} \\ \varepsilon & \text{is the strain field} \\ \mathbf{u} & \text{is the displacement field} \end{cases}$$

$\mathbf{C}$  is the stiffness tensor, which has standard properties of symmetry and positivity. Using the Einstein notations, it can be expressed as:

$$C_{ijkl} = C_{jikl} = C_{klij} \\ \exists c > 0, \quad C_{ijkl} \varepsilon_{ij} \varepsilon_{kl} > c \|\varepsilon\|^2$$

Moreover, the operators  $:$  and  $\cdot$  are defined as:

$$(\mathbf{C} : \varepsilon)_{ij} = C_{ijkl} \varepsilon_{kl} \\ (\sigma \cdot \mathbf{n})_i = \sigma_{ij} \mathbf{n}_j$$

With these hypotheses, a unique solution  $\mathbf{u} \in H^1(\Omega)$  of  $(\mathcal{P})$  exists.

## 2.3. The hourglass shape deformation process: the 2-D $(r, z)$ test case

The first test case studied is axisymmetric and results from the hourglass shape deformation process. The first advantage of this test case is that the geometry under consideration is very simple, as the computational domain is rectangular. Regular structured uniform meshes perfectly matching the real geometry can therefore be used on this case.

The contact with the pellet induced by the hourglass shape deformation is modelled by a peak in the pressure of 600  $\mu\text{m}$  around the inter-pellet plane. In view of the symmetrical conditions, only one half of the pellet is simulated (see Fig. 2).

As there is no analytical solution to this problem, a reference solution obtained with a uniform mesh matching the internal pressure gap, with a space step of  $2 \cdot 10^{-3}$  mm (2  $\mu\text{m}$ ) in each direction ( $\approx 2$  millions of DoF), was used to test the accuracy of the method.

## 2.4. The pellet cracking process: the 2-D $(r, \theta)$ test case

The second test case focuses on the pellet cracking process and respects the plane strain hypothesis. As the geometry is curved, the meshes will be regular and structured but not uniform, only "quasi uniform". The aim of this study is to test the LDC method on a less classical case. In particular, the impact of the approximation of the geometry will be examined.

The cracking process is simulated in the form of a pressure discontinuity on the internal radius of the cladding, in front of the start of the crack simulated here at 8  $\mu\text{m}$ . The pellet cracking is assumed to develop in a regular way (see [52]). Under symmetrical conditions, only 1/16 of the cladding needs to be modelled (see Fig. 3).

There exists an analytical solution to the problem set by this test case [53]. However, since this solution takes the form of a Fourier decomposition, too many terms are required to be able to reach a sufficiently accurate solution. As in the previous case, a reference solution obtained on a really fine mesh ( $\approx 2$  million DoF) with a space step of  $1 \cdot 10^{-3}$  mm in each direction adapted to the singularity, is therefore used instead of the analytical solution in the verification process.

## 2.5. The three-dimensional test case

The two previous two-dimensional test cases were combined here, giving a three-dimensional geometry. The aims of this further study are multiple: to check the accuracy of the LDC method on a three-dimensional case, to confirm whether the method is able to deal with several crossed singularities with different characteristic length-scales and to assess the performances of the local refinement strategy on a more realistic configuration.

A restrictive factor here will be the accuracy of the reference solution: the most accurate solution we have obtained so far involved a really non-uniform structured mesh with a space step ranging from 2 to 50  $\mu\text{m}$  ( $\approx 2$  millions of DoF), adapted to both singularities.

## 3. A local multi-grid algorithm

### 3.1. General principles underlying local multi-grid methods

Local multi-grid methods (e.g. [28,31–33]) are based on an inverse multi-grid process [30]. A coarse overall grid is first drawn up on the whole domain, and finer local sub-grids are set only on areas where greater precision is required. An example of hierarchical nested grids is shown in Fig. 4. Local sub-grids of this kind can be defined recursively until the required accuracy or local mesh step has been reached.

As with the standard multi-grid method, prolongation and restriction operators are defined to link up several levels of computation. The coarsest solution is then corrected thanks to an iterative process. The prolongation operator is used to transport information from a coarse grid to the next finer one. It differs from the multi-grid prolongation operator (which transports an error) and consists here in defining boundary conditions on the fine grid from the next coarser solution.

The restriction operator transports information from a fine grid to the next coarser one by adding a new source term based on the restricted fine solution. Unlike the multi-grid restriction operator,

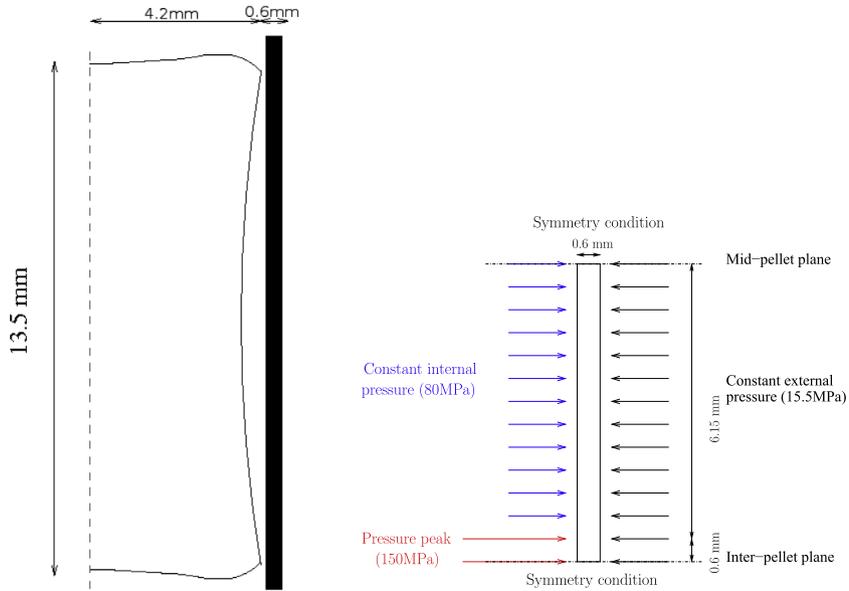


Fig. 2. Definition of the 2-D axisymmetric test case.

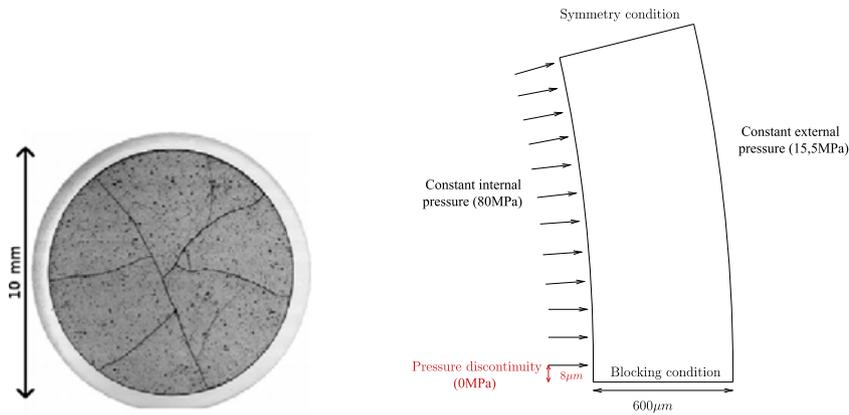


Fig. 3. Definition of the 2-D plane strain test case.

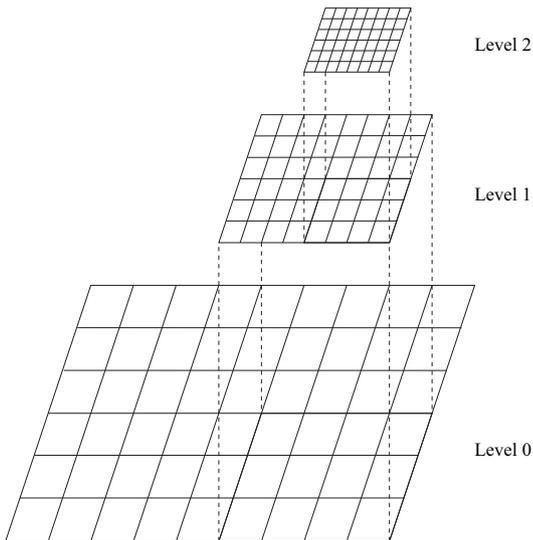


Fig. 4. Example of hierarchical nested meshes used with a local multi-grid method.

the local multi-grid restriction operator does not need to be the transpose of the prolongation operator. Several local multi-grid methods have been developed, which differ in terms of the restriction operator.

In the Local Defect Correction (LDC) method [31], the restriction step consists in correcting the coarse problem via a defect calculated on the interior nodes of the superimposed part of the grid from the next finer solution. This defect is the same as in the standard multi-grid method. This method requires a sufficiently large refinement zone. It has been applied successfully in physical contexts of several kinds (e.g. [43,46,54,55]), and its theoretical convergence has been studied in [27,48,49]. It can be used with various discretisation methods, such as Finite Difference [31,48,49], Finite Volume [28,56] and Finite Element [31,57] methods.

The Fast Adaptive Composite (FAC) method [33] consists in solving an intermediate problem on the composite grid at the restriction step. This method has been mainly applied to linear problems [58,59]. Its main advantage is that the correction affects both the fine and the coarse solutions simultaneously. In addition, it can be easily distributed on parallel architecture [58], but it gives a slower convergence in terms of number of iterations. Its main drawback is that the problem to be solved is a composite one.

The Flux Interface Correction (FIC) method [32] is based on flux conservation between the sub-grids. It was designed for being used with conservative discretisation methods such as the Finite Volume method [45,60].

In the local multi-grid methods mentioned above, the problems arising on all the grids are treated sequentially until the solution has converged on the coarsest grid. This iterative process is conventionally described by a  $\wedge$ -cycle, as in Fig. 5.

Another local multi-grid approach based on the widely used Full Multi-Grid (FMG) [50] process has been recently introduced [35,47] (see Fig. 6). The restriction operator is the same as in the LDC method, but the prolongation operator is both the standard FMG one and the standard local multi-grid one (boundary conditions).

This method can be defined as a recursive two-grid algorithm. It is attractive if the resolution on the fine sub-grids is liable to be time-consuming. When the sub-grids are localised (i.e., there are only a few DoF), a fairly accurate solution can be obtained cheaply. In this case, the benefits of the progressive smoothing process are not so obvious.

### 3.2. The local defect correction method

Let us consider the problem  $(\mathcal{P})$  defined on an open-bounded domain  $\Omega$  of boundary  $\partial\Omega$ :

$$(\mathcal{P}) : \begin{cases} \mathcal{L}(u) = f & \text{in } \Omega \\ \text{appropriate B.C.} & \text{on } \partial\Omega \end{cases} \quad (2)$$

with:

$$\begin{cases} \mathcal{L} : & \text{the operator (which is generally nonlinear)} \\ u : & \text{the solution} \\ f : & \text{the right-hand side} \end{cases}$$

The boundary conditions on  $\partial\Omega$  can be of any kind (Neumann, Dirichlet, etc.) as long as the problem is well posed.

A set of nested domains  $\Omega_l$ ,  $0 \leq l \leq l^*$  of boundary  $\partial\Omega_l$  with  $\Omega_0 = \Omega$  and  $\Omega_l \subset \Omega_{l-1}$  is then defined. The problem  $(\mathcal{P})$  is restricted to each local domain  $\Omega_l$ :

$$(\mathcal{P}_{|\Omega_l}) : \begin{cases} \mathcal{L}_{|\Omega_l}(u) = f_{|\Omega_l} & \text{in } \Omega_l \\ \text{appropriate B.C.} & \text{on } \partial\Omega_l \end{cases} \quad (3)$$

B.C. on  $\partial\Omega_l$  will be defined latter.

Each domain is discretised by a grid  $G_l$  with boundary  $\Gamma_l$  and discretisation step  $h_l < h_{l-1}$ . The nodal discrete problem associated to the problem  $(\mathcal{P}_{|\Omega_l})$  can be written without any loss of generality:

$$(\mathcal{P}_l) : \mathcal{L}_l(u_l) = f_l + g_l \quad \text{in } G_l \quad (4)$$

with:

$$\begin{cases} \mathcal{L}_l : & \text{the discrete operator associated with } \mathcal{L}_{|\Omega_l} \text{ on } G_l \\ u_l : & \text{the vector of the nodal unknowns} \\ f_l : & \text{the vector associated to the discretisation of the right-hand member } f_{|\Omega_l} \\ g_l : & \text{the vector associated to the discretisation of the B.C. on } \Gamma_l \end{cases}$$

The LDC method will work iteratively on the problem  $(\mathcal{P}_l)$  by modifying the right-hand member ( $f_l$  and  $g_l$ ). So the problem to be solved at each step (prolongation, restriction) in the  $k^{\text{th}}$  ( $k \geq 1$ )  $\wedge$ -cycle writes:

$$(\mathcal{P}_l^k) : \mathcal{L}_l(u_l^k) = f_l^k + g_l^k \quad \text{in } G_l \quad (5)$$

$f_l^k$  and  $g_l^k$  will be defined in the following sections. The B.C. ( $g_l^k$ ) remains unchanged between restriction and prolongation step at iteration  $k$ . However, as the part of the right-hand side noted  $f_l^k$  will be updated during the restriction step, for  $k \geq 1$ , we will note the difference in what follows between the right-hand side during the prolongation step  $f_{l,p}^k$  and the restriction step  $f_{l,r}^k$ .

For the sake of simplicity, all the figures below will be plotted with hierarchical meshes. However, since the equations are generic ones, they can be used with non-hierarchical meshes (see [61] for example).

#### 3.2.1. Initialisation

In what follows, we will define the initial first part of the right-hand member (see Eq. (5)) on the grid  $G_l$ , which is denoted by  $f_l^0$ , as the discretisation of the original second-hand member  $f_{|\Omega_l}$  on  $G_l$ . We also set  $f_{l,p}^1 = f_l^0$ .

The initial coarse solution  $u_0^0$  is obtained by solving  $(\mathcal{P}_0)$  which is the discrete nodal problem associated with  $(\mathcal{P})$ , defined on  $G_0$ .

The following prolongation and restriction steps will then be repeated with  $k$  ranging from 1 to  $k^*$  (see Fig. 5).

#### 3.2.2. Prolongation step: boundary conditions

The boundary conditions on the grids  $G_l$ ,  $0 < l \leq l^*$  with  $l^* > 0$  are defined as follows (see the example in Fig. 7):

- On  $\Gamma_l \cap \partial\Omega$  the boundary conditions of the original problem  $(\mathcal{P})$  are imposed.
- On  $\Gamma_l \setminus (\Gamma_l \cap \partial\Omega)$  Dirichlet boundary conditions are set. The Dirichlet values are obtained by applying the prolongation operator  $P_{l-1}^l$  to the next coarser solution  $u_{l-1}^k$ :

$$u_{\Gamma_l \setminus (\Gamma_l \cap \partial\Omega)}^k = (P_{l-1}^l(u_{l-1}^k))_{|\Gamma_l \setminus (\Gamma_l \cap \partial\Omega)}$$

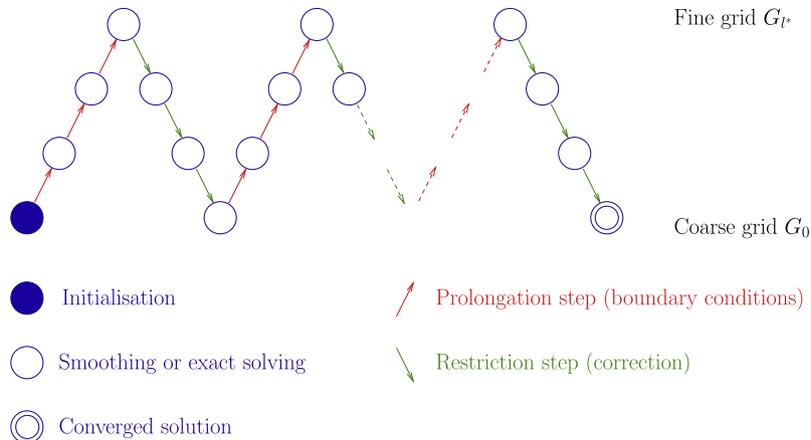


Fig. 5. Representation of a local multi-grid process:  $\wedge$ -cycle.

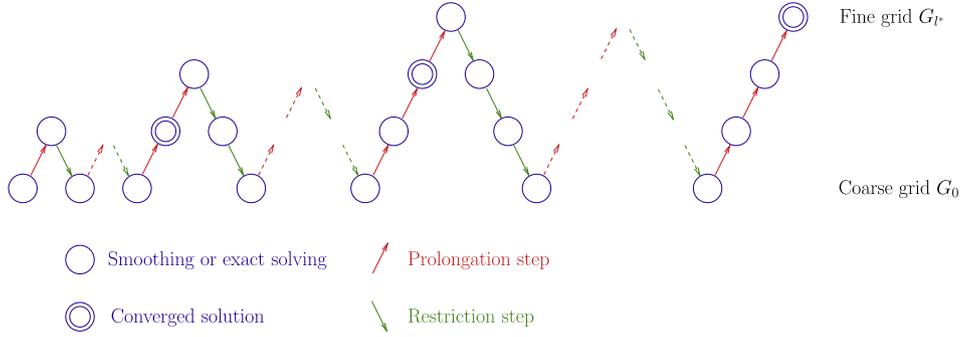


Fig. 6. Diagram of a local FMG process.

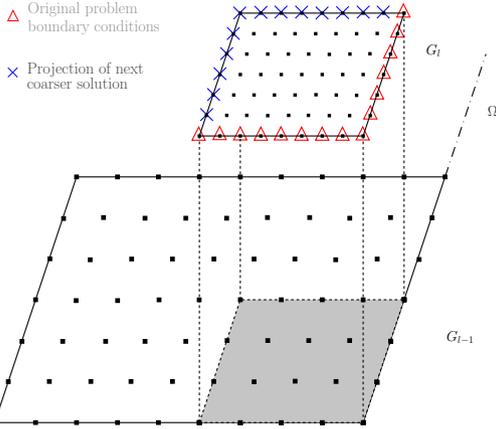


Fig. 7. Example of boundary conditions on  $G_l$  ( $l \neq 0$ ) for hierarchical meshes.

In order to solve  $(\mathcal{P}_l^k)$ , one has to also define  $f_{l,p}^k$ . As this is not the key point in the prolongation step and as the original algorithm is written for two grids, there is some disagreement in the literature about the choice of the right-hand side (RHS)  $f_{l,p}^k$  for  $1 \leq l \leq l^*$ , for  $l^* > 1$  and  $k > 1$ . There exist two main possible choices, depending on how the intermediate level is approached:

$$f_{l,p}^k = f_{l,r}^{k-1} \quad \text{or} \quad f_{l,p}^k = f_l^0$$

As the finest grid problem is solved in prolongation steps only, both choices give  $f_{l,p}^k = f_l^0$  which is consistent with the two-grid method. The impact of the choice of  $f_{l,p}^k$  at the prolongation step in the case of intermediate solving levels ( $l^* > 1$ ) will be studied in Section 4.2.

### 3.2.3. Restriction step: defect correction

The restriction step consists in correcting the coarse problem via a defect calculated from the next finer solution. The boundary conditions defined on the prolongation step for the same level are kept when solving the new problem  $(\mathcal{P}_l^k)$ .

Two sets of nodes of  $G_l$  have to be defined [31] (see Fig. 8). The set  $A_l$  contains the nodes of  $G_l$  which are strictly included in  $\Omega_l$ . This set is used to restrict the next finer solution  $u_{l+1}^k$ . The set  $\hat{A}_l$  consists of the interior nodes of  $A_l \cup (\Gamma_l \cap \partial\Omega)$  in the sense of the discretisation scheme: a node  $x$  in  $A_l \cup (\Gamma_l \cap \partial\Omega)$  is interior if  $\mathcal{L}_l(u_l)(x)$  involves only terms  $u_l(y)$  such as  $y$  is in  $A_l \cup (\Gamma_l \cap \partial\Omega)$  [31]. The set  $\hat{A}_l$  defines the zone where the RHS  $f_l^k$  will be modified.

First, the solution obtained on the fine grid  $G_{l+1}$  is restricted to the nodes of  $A_l$ :

$$\tilde{u}_l^k(x) = (R_{l+1}^l u_{l+1}^k)(x) \quad \forall x \in A_l$$

where  $R_{l+1}^l$  is an interpolation operator from the fine grid  $G_{l+1}$  to the coarse grid  $G_l$ .

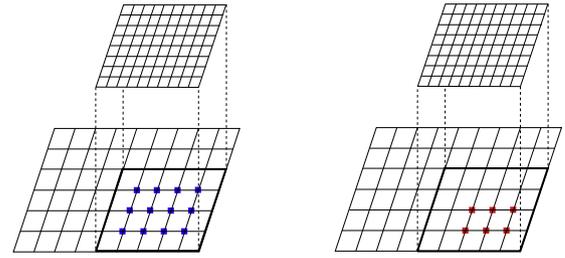


Fig. 8. Example of restriction zone  $A_l$  (on the left) and correction zone  $\hat{A}_l$  (on the right), in the case of a 5-point stencil operator and for hierarchical meshes.

The local coarse defect associated with this solution is then computed on the nodes of  $\hat{A}_l$ :

$$r_l^k(u)(x) = (\mathcal{L}_l(\tilde{u}_l^k) - f_l^0 - g_l^k)(x) \quad \forall x \in \hat{A}_l$$

Lastly, the coarse solution  $u_l^k$  is corrected by solving the coarse problem  $(\mathcal{P}_l^k)$  with the modified RHS:

$$f_{l,r}^k = f_l^0 + \chi_{\hat{A}_l} r_l^k(u) \quad (6)$$

where  $\chi_{\hat{A}_l}$  is the characteristic function of  $\hat{A}_l$ .

As the correction affects only the nodes in the subset  $\hat{A}_l$ , the LDC method needs the correction zone  $A_l$  to be large enough to be efficient [31].

The coarsest solution is then used in the next prolongation step (see Fig. 5):  $u_0^{k+1} = u_0^k$ .

Comment: Eq. (6) is based on the assumption that the defect in the fine problem  $\tilde{r}_{l+1}^k$  is negligible in comparison with  $r_l^k$ , which means that the fine resolution is sufficiently accurate (see [28] for example):

$$\|\tilde{r}_{l+1}^k\| = \|\mathcal{L}_{l+1}(u_{l+1}^k) - f_{l+1}^k - g_{l+1}^k\| \ll \|r_l^k\| \quad (7)$$

### 3.2.4. Convergence

As said in Introduction, the algorithm is proved to converge to the solution of the continuous problem [31]. Moreover, the LDC method keeps the order of convergence of the mesh but makes the error decrease [31], until the pollution error (which may be due here to the discretisation error in the non-refined part) is reached [62]. In order to obtain the optimum order of mesh convergence with respect to the local finest mesh, by extension of the results of Ferret [48], the order of the interpolation operators has to be at least the expected order of mesh convergence.

## 4. Methodology to apply the LDC method to an industrial solid mechanics test case

### 4.1. Numerical considerations

#### 4.1.1. Discretisation

The numerical resolution is performed using the standard  $Q_1$  FE method [63].

The industrial FE software CAST3M [64] developed by the CEA (French Atomic Energy Commission) is used.

#### 4.1.2. Generation of sub-grids

We decided to use hierarchical meshes, which simplify the mesh generation and the choice of prolongation and restriction operators.

The mesh size  $h_l$  of the grid  $G_l$  is defined depending on the size  $h_{l-1}$  of the previous grid  $G_{l-1}$  by:

$$h_l = \frac{h_{l-1}}{r(l)}$$

where

- $r(l)$  is the refinement ratio between the grids  $G_{l-1}$  and  $G_l$ . The same ratio is used in each direction.
- $l$  is the level of the grid  $G_l$ .

If the refinement ratio is chose to be constant for all the grids ( $r(l) = r \forall l$ ), we can express the mesh size of the grid  $G_l$  according to the initial mesh size  $h_0$  by:

$$h_l = \frac{h_0}{r^l}$$

In what follows, we have taken the same refinement ratio  $r$  for all the grids, typically 2 or 4.

#### 4.1.3. Interpolation operators

A linear interpolation (obtained from the coarse basis functions) is performed in the prolongation step, while the canonical restriction procedure (hierarchical meshes) is used in the restriction step. This choice of operators is in agreement with the expected first order accuracy of the method.

#### 4.1.4. Solver

In order to fulfil hypothesis (7), a quasi-exact resolution has to be performed on the sub-grids. As each level has a limited number of DoF, a direct solver is used to solve the problems ( $\mathcal{P}_l^k$ ). The advantage of this method is that the inverse of the stiffness tensor can be kept during the next  $\wedge$ -cycle resolutions. As the  $\wedge$ -cycles differ only in their RHS (see Section 3), the time consumption required by a LDC resolution will mostly correspond to the CPU time required to perform the first prolongation step resolutions.

#### 4.1.5. Errors

In order to assess the accuracy of the various versions of the method, several error calculations were computed. The discretisation error was then defined as the difference between the analytical solution  $\tilde{u}$  of ( $\mathcal{P}$ ) and the solution  $u_h$  of the discrete problem ( $\mathcal{P}_0^k$ ) defined on  $G_0$ .

In a real-life industrial context, the exact position of the singularity is unknown *a priori*. To be consistent with this hypothesis, the position of the singularity will be taken to be mesh dependent. This position will be chosen in a conservative way for the sake of safety. The discretisation error can therefore be divided into two terms: a modelling error and a numerical scheme error. The numerical scheme error takes the approximation of the geometry

into account. Taking  $u_d$  to denote the solution of the continuous problem with a singularity differing by  $d$  from the real one, the discretisation error can be written:

$$\|\tilde{u} - u_h\| \leq \underbrace{\|\tilde{u} - u_d\|}_{\text{modelling error}} + \underbrace{\|u_d - u_h\|}_{\text{numerical scheme error}} \quad (8)$$

In the case of a non-polyhedral open-bounded domain, even with an approximated geometry, the numerical scheme error has been proved to be still in  $O(h^2)$  [65] for the  $L^2$  norm with the standard conforming  $Q_1$  FE method. It has been established in [66] that the modelling error is of the first order with respect to  $d$ . In practice, since  $d$  may vary as  $h$ , the mesh convergence of the discretisation error is then in  $O(dh)$ . Using a local multi-grid process, it is proposed here to reduce the discretisation error by adding local meshes with increasingly fine space steps up to the finest local mesh size  $h_{fine}$ . The LDC method can be expected to converge as  $O(d_{h_{fine}})$  [48], until stagnated to the pollution error, see Section 3.2.4. In our case, the pollution error may be due to either the discretisation error in the non-refined part or the coarse approximation of the geometry (hierarchical meshes).

Two error norms will be examined: the  $L^2$  norm and the  $L^\infty$  norm (or maximum norm). These two norms are complementary as the  $L^2$  error norm gives information about the response of the whole structure whereas the  $L^\infty$  error norm provides more localised information. The discrete  $L^2$  norm on an approximate domain  $\Omega_h$  can be written as follows:

$$\|\varphi_h\|_{L^2(\Omega_h)}^2 = \sum_{K \subset \Omega_h} \|\varphi_h\|_{L^2(K)}^2$$

where  $K$  is an element of the triangularisation of  $\Omega_h$ .  $\|\varphi_h\|_{L^2(K)}^2$  was performed here using a numerical integration procedure, which is exact on  $Q_1(K)$  ( $R(\varphi) \equiv 0$  if  $\varphi \in Q_1$ ):

$$\|\varphi_h\|_{L^2(K)}^2 = \int_K \varphi_h^2 dx = \sum_{i=1}^{n_K} \frac{\text{meas}(K)}{n_K} \varphi_h^2(x_i) + R(\varphi_h^2)$$

The subscript  $i$  denotes a vertex of the element  $K$  and  $n_K$  stands for the number of vertices in  $K$ .

The relative discrete  $L^2$  error norm is the ratio of the absolute discrete  $L^2$  error norm to the discrete  $L^2$  norm of the reference solution  $\tilde{u}$ :

$$\|e_h\|_{L^2(\Omega_h)} = \frac{\|\tilde{u} - u_h\|_{L^2(\Omega_h)}}{\|\tilde{u}\|_{L^2(\Omega_h)}}$$

The relative discrete  $L^\infty$  error norm is defined as follows:

$$\|e_h\|_{L^\infty(\Omega_h)} = \frac{\max_{\Omega_h} |\tilde{u} - u_h|}{\max_{\Omega_h} |\tilde{u}|}$$

In addition, as we are interested in fine local values, we also compute composite error norms. On an initial coarse mesh, these composite error norms can be used to take the approximate solutions obtained on fine local nodes into account. In this case, the relative error norms are summed recursively, working from the finest grid to the coarsest one. For example, for the  $L^2$  error norm, we have:

$$\|e_h\|_{L^2, \text{comp}}^2 = \sum_{i=l'}^0 \|e_h\|_{L^2(\Omega_{h,i} \setminus \Omega_{h,i+1})}^2$$

We will use the notation  $\|e_h\|_{L^2, \text{comp}}$  (respectively  $\|e_h\|_{L^\infty, \text{comp}}$ ) to write the composite relative  $L^2$  error norm (respectively the composite relative  $L^\infty$  error norm).

In practice, as announced in Section 2.1, the reference solution adopted for this study was the discrete solution  $\tilde{u}_h$  of ( $\mathcal{P}$ ) obtained with a classical  $Q_1$  FE resolution on a very fine mesh  $\tilde{G}_0$ .

Comment: In the case of curved boundaries which can be discretised differently on  $G_l$  and  $\tilde{G}_0$ , all the nodes of  $G_l$  have to be included in  $\tilde{G}_0$  in order to interpolate  $\tilde{u}_h$  on  $G_l$  thanks to the shape functions of  $\tilde{G}_0$ . For this purpose, the nodes of  $G_l$  which are exterior to  $\tilde{G}_0$  has then orthogonally projected on  $\tilde{G}_0$ .

#### 4.1.6. Convergence of the $\wedge$ -cycles

The convergence of the  $\wedge$ -cycles have been tested by comparing two successive coarse solutions:

$$\frac{\|u_0^k - u_0^{k-1}\|_{L^2(G_0)}}{\|u_0^k\|_{L^2(G_0)}} \leq 1.10^{-5}$$

#### 4.2. Prolongation operator at the intermediate level

As seen in Section 3, the LDC algorithm is classically written for two grids. It can be easily extended to multiple grids, by generating sub-problems recursively.

However, some indecision occurs when one has to define the RHS of the problem ( $\mathcal{P}_l^k$ ) at prolongation steps on the intermediate levels ( $0 < l < l^*$ ). This level can be regarded as either a fine level for the next coarser one or a coarse level from which a finer grid is defined.

It is usually assumed that each intermediate level is a local coarse level from which a new two-grid algorithm is generated. The RHS of the previous restriction step is therefore used at the next prolongation step:  $f_{l,p}^k = f_{l,r}^{k-1}$  (see [49] for example).

Another point of view consists in drawing up an overall picture of the multi-grid cycle (cf. Fig. 5). At the prolongation step, the intermediate levels therefore play the role of fine grids, while at the restriction step they serve as coarse grids. The RHS  $f_{l,p}^k$  can therefore be set at  $f_l^0$  during the prolongation step.

These two possibilities are tested on the two-dimensional axisymmetric test case presented in Section 2.3.

In this section, the refinement ratio  $r$  is set at 2. An example of nested sub-grids is presented in Fig. 9(b), where the initial coarse mesh step is  $h_0 = h_i = 327 \mu\text{m}$ . Remember that the pressure gap is located  $600 \mu\text{m}$  from the bottom of the cladding and that it is not set *a priori* on a discretisation point. Sub-grids are generated *a priori*, based on the relative error in the displacement obtained from the reference solution (see Fig. 9(a)). The meshes are localised around the singularity in the whole thickness of the cladding.

As was to be expected, the finest grid amounts to only a very small part of the initial domain. Since the hierarchical meshes are taken here to be structured and uniform (Cartesian meshes), fast solvers can be used on each mesh.

Fig. 10 gives the relative discrete  $L^2$  error norm with respect to the distance between the real singularity and its approximation on the initial coarse mesh, and with respect to the number of sub-grids. The “fine like” prolongation operator refers to the choice  $f_{l,p}^k = f_l^0$  whereas the “coarse like” prolongation operator refers to the choice  $f_{l,p}^k = f_{l,r}^{k-1}$ .

The first conclusion to be drawn here is that, as was to be expected, the mesh convergence of the overall structured uniform

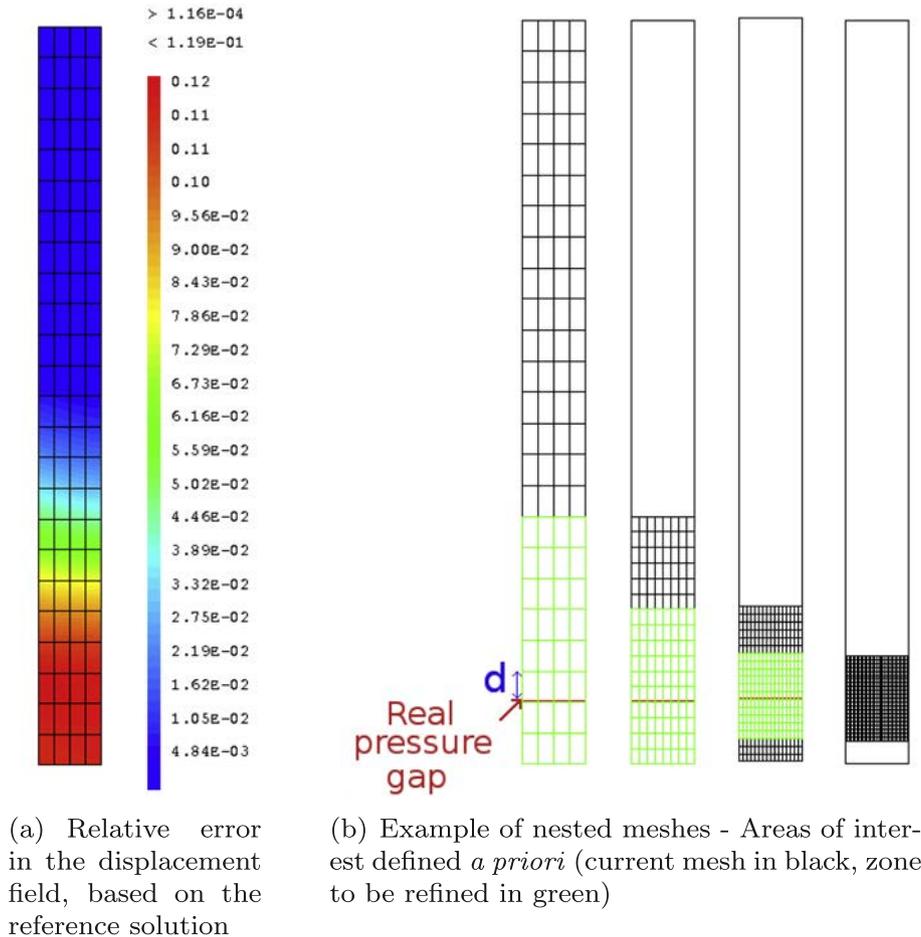
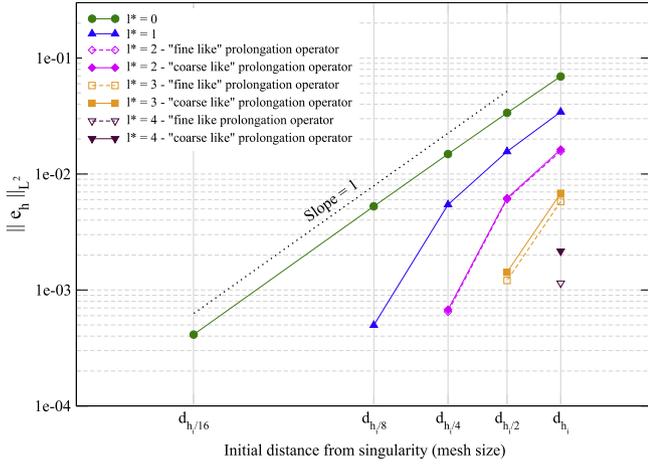


Fig. 9. The 2-D axisymmetric test case –  $h_0 = h_i$ .

LDC method applied to the 2D axisymmetric model  
Influence of RHS on the prolongation step



**Fig. 10.** Influence of the prolongation operator on the intermediate levels – Relative  $L^2$  error norm – The 2-D ( $r,z$ ) test case.

meshes is of the first order with respect to the distance from singularity.

The second conclusion is that with both approaches to the prolongation operator, the order of convergence with respect to the local finest distance to the singularity is maintained until stagnation occurs: the same error level is obtained with a local refinement as with an overall mono-grid with a discretisation step equal to the finest local one. The LDC method therefore converges as  $O(d_{h_{fine}})$ , where  $d_{h_{fine}}$  corresponds to the local distance from the singularity.

These two approaches yield equivalent results before reaching the pollution error due here to the approximation error in the non-refined part. The “fine like” prolongation operator seems to be a little more accurate. However, both hypotheses take the same number of  $\wedge$ -cycles to converge, which means that they both require a similar CPU time.

The relative maximum error norms are given in Table 1.

We can observe that the optimum convergence in  $O(d_{h_{fine}})$  is also reached before slowing down to deal with fine error levels ( $< 1 \cdot 10^{-3}$ ). For example, the same error was obtained with  $h_0 = h_i/4$  and  $l^* = 0$  as with  $h_0 = h_i/2$  and  $l^* = 1$  or  $h_0 = h_i$  and  $l^* = 2$ . The behaviour of the norms  $L^2$  and  $L^\infty$  was found to be similar. Here again, the “fine like” prolongation operator gives a slightly more accurate solution at the finest error levels.

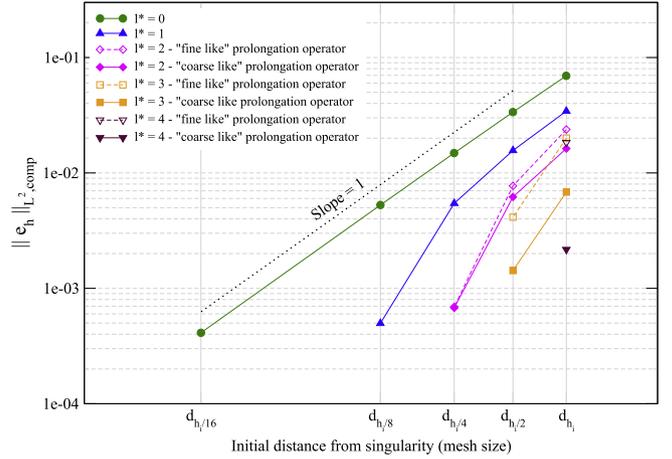
In order to also compare the solution obtained on the sub-grids using these two approaches, the composite error norms were then examined. The relative composite  $L^2$  error norms are plotted in Fig. 11 and the relative composite  $L^\infty$  error norms are given in Table 2.

**Table 1**

Comparison between “fine like” (‘f’) and “coarse like” (‘c’) prolongation operators – The 2-D ( $r,z$ ) test case – Relative maximum error norm.

Relative maximum error norm depending on the prolongation operator					
$h_0$	$l^*$				
	0	1	2	3	4
$h_i$	$1.14 \cdot 10^{-1}$	$5.83 \cdot 10^{-2}$	‘f’: $2.69 \cdot 10^{-2}$ ‘c’: $2.78 \cdot 10^{-2}$	‘f’: $9.57 \cdot 10^{-3}$ ‘c’: $1.13 \cdot 10^{-2}$	‘f’: $1.73 \cdot 10^{-3}$ ‘c’: $2.73 \cdot 10^{-3}$
$h_i/2$	$5.77 \cdot 10^{-2}$	$2.72 \cdot 10^{-2}$	‘f’: $1.05 \cdot 10^{-2}$ ‘c’: $1.07 \cdot 10^{-2}$	‘f’: $1.82 \cdot 10^{-3}$ ‘c’: $2.18 \cdot 10^{-3}$	
$h_i/4$	$2.62 \cdot 10^{-2}$	$9.71 \cdot 10^{-3}$	‘f’: $1.17 \cdot 10^{-3}$ ‘c’: $1.22 \cdot 10^{-3}$		
$h_i/8$	$9.65 \cdot 10^{-3}$	$9.42 \cdot 10^{-4}$			
$h_i/16$	$7.97 \cdot 10^{-4}$				

LDC method applied to the 2D axisymmetric model  
Influence of RHS on the prolongation step



**Fig. 11.** Influence of the prolongation operator on the intermediate level - Relative  $L^2$  composite error norm – The 2-D ( $r,z$ ) test case.

Contrary to the previous results, these results show the existence of a considerable difference between the effects of the two types of prolongation operator. When the “coarse like” prolongation operator was used, the convergence remained in  $O(d_{h_{fine}})$  with both norms except for the finest error level. These results are consistent with those obtained by Anthonissen et al. [55]. The accuracy achieved with this operator is therefore homogeneous from the finest to the coarsest grid. On the other hand, an early stagnation appears with the “fine like” prolongation operator. The coarser the initial mesh is, the earlier the stagnation appears in the refinement process. In this case, the results obtained therefore differed considerably from the expected ones.

In conclusion, both prolongation operators accurately correct the initial coarse solution (right efforts transmission) but only the “coarse like” operator gives accurate solutions on fine local grids.

As we are interested in local values, the “coarse like” ( $f_{L,p}^k = f_L^{k-1}$ ) prolongation operator will therefore be consistently used in what follows.

#### 4.3. A posteriori error estimation

This section focuses on how to automate the detection of areas of interest, which was useful for two reasons. First, in the previous section the pollution error was assumed to be due to the discretisation error in the non-refined part. It was therefore proposed to determine optimum refined zones in order to reach the required

**Table 2**

Comparison between “fine like” (‘f’) and “coarse like” (‘c’) prolongation operators – The 2-D (r,z) test case – Relative maximum composite error norm.

Relative composite maximum error norm depending on the prolongation operator					
$h_0$	$l^*$				
	0	1	2	3	4
$h_i$	$1.14 \cdot 10^{-1}$	$5.83 \cdot 10^{-2}$	‘f’: $4.16 \cdot 10^{-2}$ ‘c’: $2.78 \cdot 10^{-2}$	‘f’: $3.83 \cdot 10^{-2}$ ‘c’: $1.15 \cdot 10^{-2}$	‘f’: $3.67 \cdot 10^{-2}$ ‘c’: $2.79 \cdot 10^{-3}$
$h_i/2$	$5.77 \cdot 10^{-2}$	$2.72 \cdot 10^{-2}$	‘f’: $1.65 \cdot 10^{-2}$ ‘c’: $1.10 \cdot 10^{-2}$	‘f’: $1.12 \cdot 10^{-2}$ ‘c’: $2.23 \cdot 10^{-3}$	
$h_i/4$	$2.62 \cdot 10^{-2}$	$9.94 \cdot 10^{-3}$	‘f’: $1.44 \cdot 10^{-3}$ ‘c’: $1.24 \cdot 10^{-3}$		
$h_i/8$	$9.65 \cdot 10^{-3}$	$9.62 \cdot 10^{-4}$			
$h_i/16$	$7.97 \cdot 10^{-4}$				

precision (i.e., to obtain mono-grid errors with the same finest mesh step). Secondly, since we wanted to be able to move singularities, automatically adjusting refinement zones were required.

We decided to use the Zienkiewicz and Zhu (ZZ) *a posteriori* error estimator [40]. This is the most commonly used *a posteriori* error estimator (see for example [34,67–70]), and its super-convergent patch recovery version [71,72] gives excellent performances in terms of its accuracy versus time consumption. In addition, it is easy to use and is already available in some industrial codes, as Code\_Aster [73] or CAST3M [64]. This estimator has been proved theoretically to be valid for dealing with linear one-dimensional problems [72].

#### 4.3.1. Presentation of Zienkiewicz and Zhua posteriori error estimator

The ZZ *a posteriori* error estimator [40,71,72] consists in constructing a stress solution  $\sigma_h^*$  which is smoother than the FE one  $\sigma_h$ . The local estimator  $\eta_{E,h}$  on an element  $E$  is defined as:

$$\eta_{E,h} = \|(\sigma_h^*)_E - (\sigma_h)_E\| \quad (9)$$

The value in the element is obtained from the node values. With the FE stress  $\sigma_h$ , the values at the discretisation nodes are obtained by interpolating the values from the Gaussian points. To obtain the estimated stress  $\sigma_h^*$ , two methods have been proposed by Zienkiewicz and Zhu:

- The simplest and cheapest one [40] consists in averaging the value of the FE stress  $\sigma_h$  on the elements surrounding the node. However, this method is not very efficient in the case of very coarse grids and high-order polynomial bases.
- The second one, which is called the “super convergent patch recovery” method [71,72], is based on the use of patches, which are a combination of several elements. In each patch, a high-order polynomial function is defined, which minimises the root mean square gap with respect to  $\sigma_h$  at so-called “super convergent” points (Gaussian points in one dimensional problems). The average value of each patch contribution is then calculated in order to obtain  $\sigma_h^*$ . This method is a little bit more time-consuming but yields better approximations.

In order to generate LDC grids automatically in the maximum error zones, we decided to apply the ZZ *a posteriori* error estimator recursively during the first prolongation step to obtain each next finer grid. At each level  $l$ , we therefore refine the elements  $L$  in the grid  $G_l$  at which the local ZZ error  $e$  satisfies:

$$e_L > \alpha \left( \max_{K \subset G_l} e_K - \min_{K \subset G_l} e_K \right) \quad (10)$$

where

$$0 \leq \alpha \leq 1, \quad \alpha \text{ a given constant}$$

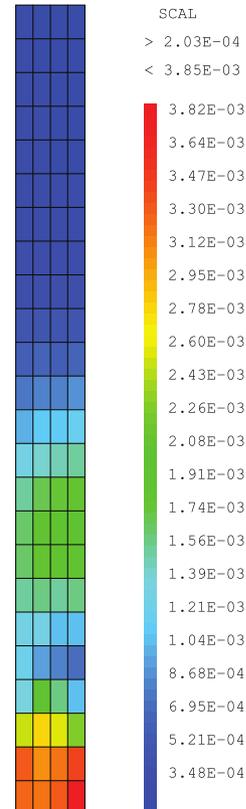
We define the minimum number of elements required to build a new local sub-grid (typically  $2^D$  where  $D$  is the spacial dimension). Given the definition of  $\hat{A}_l$  and  $A_l$  in Section 3.2.3, we also decided to enlarge the refinement zone in order to keep all the elements detected in the restriction zone  $\hat{A}_l$ .

The smaller  $\alpha$  is, the larger the number of elements to be refined will be. In what follows, we will look for the optimum value of  $\alpha$ , that is to say the largest value of  $\alpha$  (i.e., the smallest number of elements) giving the predicted error level.

With this strategy, fewer refinement levels are required than with a two-refinement comparison error indicator [47,74] where the level  $l$  is finally obtained thanks to the level  $l - 2$ .

#### 4.3.2. The two-dimensional axisymmetric test case

The local error field calculated using the ZZ estimator is plotted in Fig. 12 in the case of an initial coarse mesh of uniform cells with the mesh step  $h_i = 327 \mu\text{m}$ .



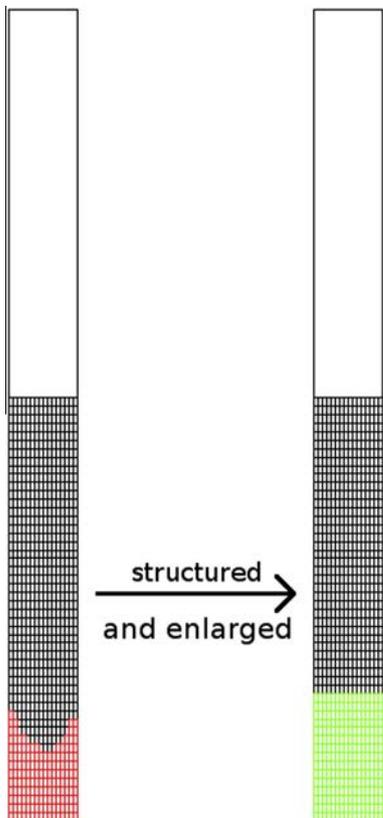
**Fig. 12.** Local error estimated using the ZZ *a posteriori* error estimator – The 2-D axisymmetric test case.

Upon applying the detection formula, we noted that the automatically detected areas (Fig. 13, left) are not structured meshes, contrary to the initial mesh. In order to keep the advantages of regular and structured meshes (such as the natural tensor formulation, good convergence properties, etc.), we decided to add some elements to each mesh to make them structured (Fig. 13, right) [29,55]. In practice, the detected zone is automatically embedded in a rectangle, then enlarged to ensure that all the detected elements will be in the correction zone  $\hat{A}_i$ . If several zones are detected, each of them is treated separately. This structuring algorithm may be not optimum in terms of DoF if the areas detected are really misshapen, which rarely occurs in the case of local singularities.

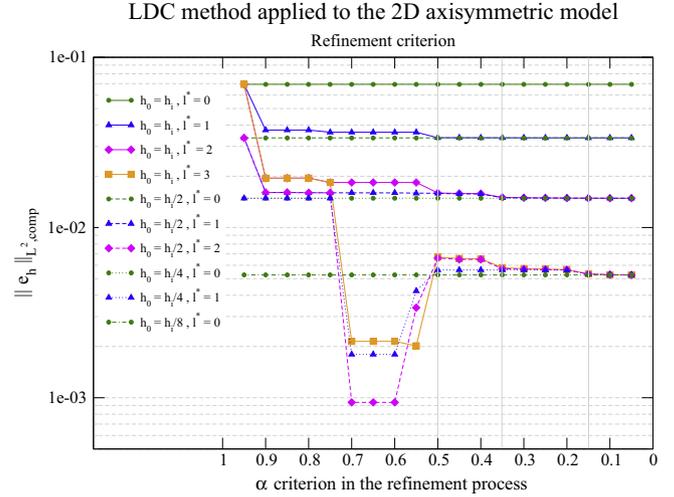
Fig. 14 gives the composite  $L^2$  error of the LDC method with respect to the refinement criterion  $\alpha$  chosen for use with the ZZ automatic detection method. The initial mesh step  $h_i$  is the same as in Section 4.2 ( $h_i = 327 \mu\text{m}$ ).

First we can see from this figure that the errors converge at the expected values. In addition, there exists an optimum refinement criterion  $\alpha$  for each expected error, which is independent of the number of sub-grids used and the initial mesh size. A single  $\alpha$  depending only on the prescribed accuracy will therefore be used at all the refinement levels. However, no simple correlation between the optimum  $\alpha$  and the expected error level was found to exist.

It is worth noting that at some error levels, large values of  $\alpha$  lead to a smaller error in the local refinements than in the overall ones. This pattern has been previously reported in the literature (see [28] for example). However, these values will not be used here since they do not yield the asymptotic error value.



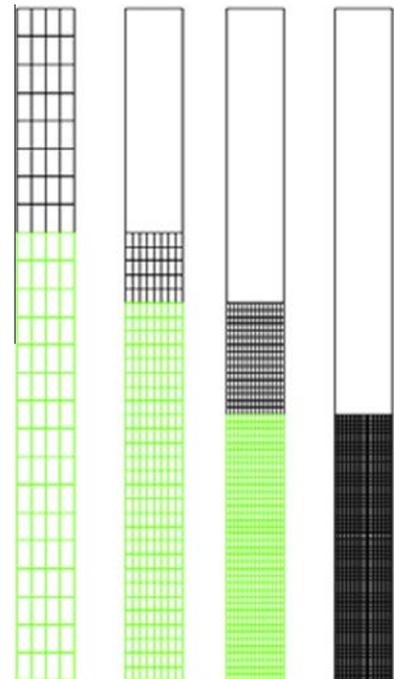
**Fig. 13.** Example of an area detected (in red) and the changes made in order to obtain a structured mesh (in green). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)



**Fig. 14.** Composite  $L^2$  error norm, depending on the  $\alpha$  criterion used on the refinement process – The 2-D ( $r, z$ ) test case.

Using the optimum value of  $\alpha$  obtained, the refinement zones were larger than in the *a priori* study: see for example Figs. 15 and 9(b), featuring the same initial mesh. This finding confirms that the stagnation previously observed resulted from a refinement area not being sufficiently large to obtain the required accuracy.

The optimum meshes (see Fig. 15) obtained seem to suggest that the effect of singularities is not very localised. However, if we look more closely at Fig. 14, we can see that the asymptotic convergence was very slow. In an industrial context, one could possibly be less strict. For example, allowing an extra margin of 10% in the required accuracy would mean that up to 50% of the refined elements could be saved.



**Fig. 15.** An example of nested meshes – *A posteriori* error estimation with  $\alpha = 0.15$  – The 2-D axisymmetric test case (current mesh in black, zone to be refined in green) –  $h_0 = h_i$ . (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

**Table 3**  
Comparison between “fine like” (‘f’) and “coarse like” (‘c’) prolongation operators – The 2-D ( $r, z$ ) test case with ZZ optimum refinement criterion  $\alpha$  – Relative maximum composite error norm.

Relative composite maximum error norm, depending on the prolongation operator					
$h_0$	$l^*$				
	0	1	2	3	4
$h_i$	$1.14 \cdot 10^{-1}$	$5.78 \cdot 10^{-2}$	‘f’: $2.83 \cdot 10^{-2}$ ‘c’: $2.63 \cdot 10^{-2}$	‘f’: $1.10 \cdot 10^{-2}$ ‘c’: $9.66 \cdot 10^{-3}$	‘f’: $2.04 \cdot 10^{-3}$ ‘c’: $7.95 \cdot 10^{-4}$
$h_i/2$	$5.77 \cdot 10^{-2}$	$2.62 \cdot 10^{-2}$	‘f’: $1.05 \cdot 10^{-2}$ ‘c’: $9.67 \cdot 10^{-3}$	‘f’: $2.04 \cdot 10^{-3}$ ‘c’: $7.95 \cdot 10^{-4}$	
$h_i/4$	$2.62 \cdot 10^{-2}$	$9.67 \cdot 10^{-3}$	‘f’: $1.08 \cdot 10^{-3}$ ‘c’: $7.95 \cdot 10^{-4}$		
$h_i/8$	$9.65 \cdot 10^{-3}$	$8.03 \cdot 10^{-4}$			
$h_i/16$	$7.97 \cdot 10^{-4}$				

The mesh convergence obtained with the LDC method using an optimum criterion is presented in Table 3. Here again, the two prolongation operators described in Section 4.2 are compared to make sure that the previous conclusions are still valid in the case of optimum refinement zones.

The composite maximum error norm began to stagnate with the “fine like” prolongation operator, while the optimum accuracy in  $O(d_{h_{fine}})$  was obtained with the “coarse like” prolongation operator. A suitable choice of refinement zones therefore has beneficial effects on both versions of the LDC method. However, the “fine like” operator still does not completely correct the solution on the fine levels. In conclusion, the “fine like” version seems to always stagnate sooner than the “coarse like” version from the composite error norm point of view.

#### 4.3.3. The two-dimensional plane strain test case

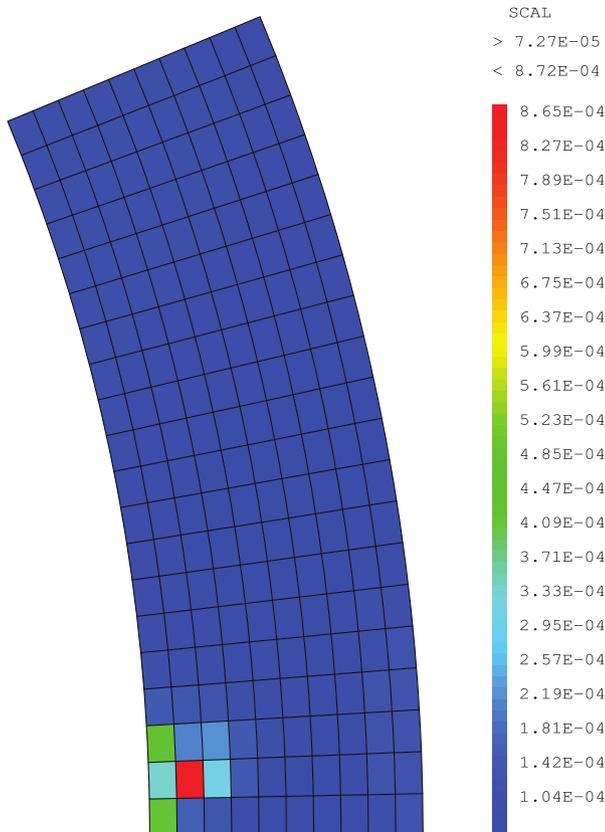
We then conducted a similar study on the 2-D plane strain test case (see Section 2.4). The use of a hierarchical local method of mesh generation means that the approximation of the curvature is still the coarsest one.

In this case, the maximum error zone obtained using the ZZ *a posteriori* error estimator was much more localised, even with a coarse initial mesh (see Fig. 16). This localisation can be explained by the fact that the structure was three times less stretched than in the previous study.

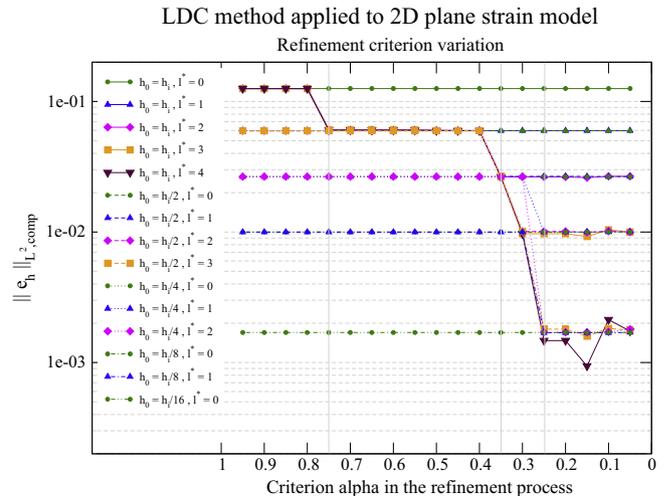
As the zone with high errors is very restricted and as a minimum number of elements detected is required to create a new level, the refinement criterion orders the number of sub-grids generated, except with very fine meshes. The influence of the refinement criterion on the  $L^2$  composite error norm is shown in Fig. 17. Here the initial mesh step  $h_i$  was  $218 \mu\text{m}$ .

As previously, an optimum criterion can always be set, depending only on the required error level. However, the optimum values obtained per error level differ from those obtained in the previous test case (see Fig. 14). The optimum values of  $\alpha$  therefore seem to also depend on the effects of the singularity under study. Results of the mesh convergence study on the maximum composite error norm with optimum  $\alpha$  are given in Table 4 in the case of both prolongation operators. An example of optimum nested sub-grids is given in Fig. 18.

With the “coarse like” operator, the method still converges as  $O(d_{h_{fine}})$ , which is the expected order of convergence. This confirms



**Fig. 16.** Local error given by the ZZ *a posteriori* error estimator – The 2-D plane strain test case.



**Fig. 17.** Composite  $L^2$  error norm depending on the  $\alpha$  criterion used in the refinement process – The 2-D ( $r, \theta$ ) test case.

that the geometrical discretisation error is negligible in comparison with the modelling error (see section 4.1).

As in the 2-D (r,z) test case, an early stagnation occurs with the “fine like” prolongation operator, especially with the initial coarsest mesh ( $h_0 = h_i$ ). The error on the composite LDC grid was greatly deteriorated. For example, the error obtained in a simulation performed with  $h_0 = h_i$  and  $l^* = 4$  with the “fine like” operator was 40 times greater than with the “coarse like” operator which leads to the expected error.

#### 4.3.4. Conclusions

In conclusion, with both test cases, the refinement process was successfully automated using the ZZ *a posteriori* error estimator. The performances of the combined LDC and ZZ methods will be studied in Section 5 in terms of the CPU time and the memory space. As with all *a posteriori* error estimators, a criterion serving as a threshold has to be set. As we have seen, this criterion contributes importantly to reaching the optimum convergence. Although this study has shown that for each of the cases tested, the optimum criterion depended only on the error level imposed, its value varies with the test case under investigation. No simple correlations have been found to exist so far between the optimum criterion and the error level imposed with any of the singularities under consideration. However, in both studies, a ZZ criterion of 0.25 seems to constitute a good compromise between the accuracy and the number of elements refined.

In what follows, the ZZ *a posteriori* error estimator will be used consistently to generate the sub-grids automatically. Optimum or compromise values of the refinement criterion  $\alpha$  will alternatively be set.

Moreover, this study confirmed that the “coarse like” prolongation operator is always more accurate than the “fine like” operator. The prolongation operator used will always be the “coarse like” one.

#### 4.4. Extension to the three-dimensional test case

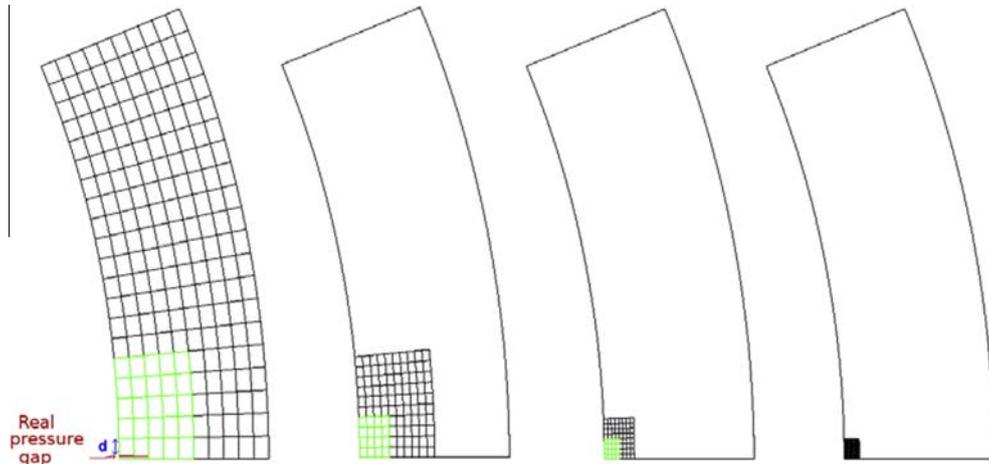
As stated above (see Section 2.5), the two previous 2-D cases are combined here in a three-dimensional geometry.

It is not easy to choose the optimum refinement criterion  $\alpha$  *a priori*. As we saw in Section 4.3, this parameter depends on the test case under investigation. In addition, using the minimum value of the two criteria obtained in the previous 2-D studies does not yield the expected results: it is too large in the case of the largest errors and too small in that of the smallest ones. A further convergence study depending on the  $\alpha$  criterion therefore needs to be performed. As this sensitivity study is very costly, especially in the 3-D context, in view of the conclusions of Section 4.3, we decided to conduct the mesh convergence study with a fixed criterion  $\alpha = 0.25$ , which seemed to be a good compromise for the two 2-D studies.

**Table 4**

Comparison between the results obtained with the “fine like” (‘f’) and “coarse like” (‘c’) prolongation operators - The 2-D (r,θ) test case with optimum refinement criterion  $\alpha$  - Relative maximum composite error norm

Relative composite maximum error norm depending on the prolongation operator					
$h_0$	$l^*$				
	0	1	2	3	4
$h_i$	$1.75 \cdot 10^{-1}$	$8.67 \cdot 10^{-2}$	‘f’: $8.40 \cdot 10^{-2}$ ‘c’: $3.91 \cdot 10^{-2}$	‘f’: $8.29 \cdot 10^{-2}$ ‘c’: $1.47 \cdot 10^{-2}$	‘f’: $8.24 \cdot 10^{-2}$ ‘c’: $2.40 \cdot 10^{-3}$
$h_i/2$	$8.61 \cdot 10^{-2}$	$3.94 \cdot 10^{-2}$	‘f’: $1.59 \cdot 10^{-2}$ ‘c’: $1.53 \cdot 10^{-2}$	‘f’: $4.15 \cdot 10^{-3}$ ‘c’: $2.82 \cdot 10^{-3}$	
$h_i/4$	$3.93 \cdot 10^{-2}$	$1.52 \cdot 10^{-2}$	‘f’: $3.03 \cdot 10^{-3}$ ‘c’: $2.70 \cdot 10^{-3}$		
$h_i/8$	$1.52 \cdot 10^{-2}$	$2.68 \cdot 10^{-3}$			
$h_i/16$	$2.70 \cdot 10^{-3}$				



**Fig. 18.** Example of nested meshes - *A posteriori* error estimation with an optimum  $\alpha = 0.25$  - The 2-D plane strain test case (current mesh in black, zone to be refined in green) -  $h_0 = h_i/2$ . (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

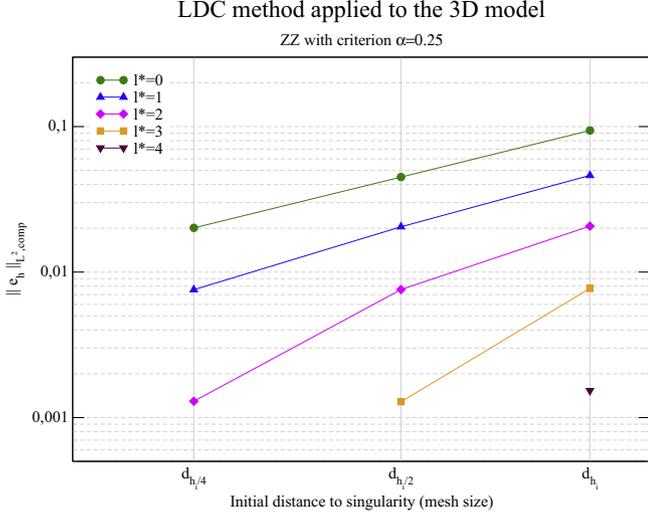


Fig. 19. Composite  $L^2$  error norm depending on the mesh – The 3-D test case.

The composite  $L^2$  error norm between the reference solution and the LDC one is plotted in Fig. 19. In this case,  $d$  denotes the initial distance between the real position of the intersection between the two singularities and its mesh approximation.

The first-order convergence required was reached in the case of uniform mono-grid solutions. However, more accurate mono-grid solutions could not be obtained due to the large number of DoF involved.

The error improvement obtained using the local multi-grid refinement strategy turned out to be still possible in the three-dimensional context, since more accurate solutions could easily be obtained. The LDC method applied to two crossed three-dimensional singularities converges as  $O(d_{h_{fine}})$ , since the errors obtained in this case were quasi optimum. The results obtained here with a non-user-dependent choice of  $\alpha$  are really acceptable. They confirms the validity of the *a priori* choice of the value of  $\alpha$  (0.25).

An example of refined meshes obtained with  $\alpha = 0.25$  is given in Fig. 20. The ZZ *a posteriori* error estimator automatically detected the crossed singularities, and the finest meshes were highly localised around the two singularities. The meshes were obtained in almost the same way as in the 2-D cases. The 2-D detection algorithm was applied on each horizontal layer. The different structured layers are regrouped according to their relative size. However, a more generic algorithm (such as a shape detection method) could be also used. Actually, as the LDC is user-adjustable, it can be combined with any detection algorithm.

As previously, the composite maximum relative error norm was also studied, and the results are given in Table 5.

Here again, the composite  $L^2$  and the  $L^\infty$  error norms showed the same behaviour, and convergence of the composite  $L^\infty$  error norm was reached in  $O(d_{h_{fine}})$ . This LDC approach combined with the ZZ *a posteriori* error estimator seems to constitute a really promising strategy, since for example, the maximum error norm was decreased by as much as 100-fold by adding 3 local levels of sub-grids to an initially uniform mesh with the space step  $h_0 = h_i$ .

## 5. Performances of the LDC method in an engineering context

### 5.1. Advantages of using of sub-grids

We have previously established that a similar level of accuracy can be achieved with a uniform fine mono-grid mesh as with a global coarse mesh and several local sub-grids, as long as the finest local mesh size is the same in each case. To optimise the trade-off

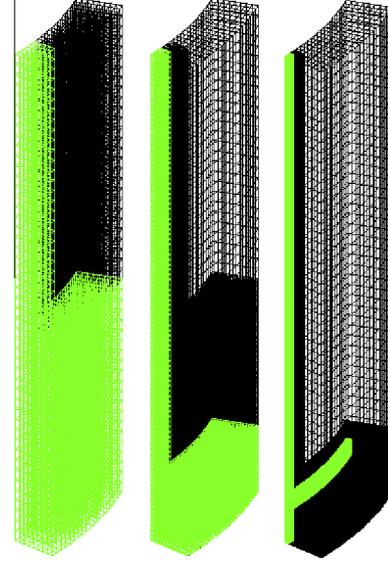


Fig. 20. Example of optimum hierarchical meshes – ZZ *a posteriori* error estimator with  $\alpha = 0.25$  – The 3-D test case –  $h_0 = h_i/2$ .

Table 5

The 3-D test case with refinement criterion  $\alpha = 0.25$  - Relative maximum composite error norm

$h_0$	$\ e_h\ _{L^\infty, comp}$ with $\alpha = 0.25$					
	$l^*$	0	1	2	3	4
$h_i$		$1.15 \cdot 10^{-1}$	$5.75 \cdot 10^{-2}$	$2.61 \cdot 10^{-2}$	$9.99 \cdot 10^{-3}$	$2.12 \cdot 10^{-3}$
$h_i/2$		$5.64 \cdot 10^{-2}$	$2.60 \cdot 10^{-2}$	$1.00 \cdot 10^{-2}$	$1.91 \cdot 10^{-3}$	
$h_i/4$		$2.57 \cdot 10^{-2}$	$9.93 \cdot 10^{-3}$	$1.80 \cdot 10^{-3}$		

between the accuracy and the means (CPU time, memory space) required, we studied the behaviour of the composite  $L^2$  error norm in terms of the CPU time (see Fig. 21 for the 2-D  $(r, \theta)$  test case) as well as in terms of the total number of nodes (see Fig. 22 for the 2-D  $(r, \theta)$  test case). In both studies, the refinement process was performed with the optimum value of the refinement criterion  $\alpha$  obtained in section 4.3.

From these figures, it can be concluded that the greater the accuracy required is, the more advantageous the use of an initial coarse mesh with many suitable sub-grids will be, in terms of both the CPU time and the memory space required. These results are consistent with those obtained by Ramière [60] with the FIC method.

Similar conclusions can be drawn as regards the 2-D  $(r, z)$  and 3-D test cases.

### 5.2. Refinement ratio study

The LDC method with a FE discretisation is classically applied with a refinement ratio of 2 (e.g. [31,43,55,75]). In some studies in which the LDC method was combined with a Finite Volume discretisation procedure, a refinement ratio of 3 was used (e.g. [46]) in order to simplify the projections, but very few studies [28] have involved comparisons between several refinement ratios on the same test case. As there are no theoretical limitations to the choice of refinement ratio, we decided to compare the performances of the LDC method between a refinement ratio of 2 and 4. The results for the 2-D  $(r, \theta)$  test case are presented in Table 6.

As predicted theoretically, similar error levels were obtained in simulations performed with two successive refinements with a

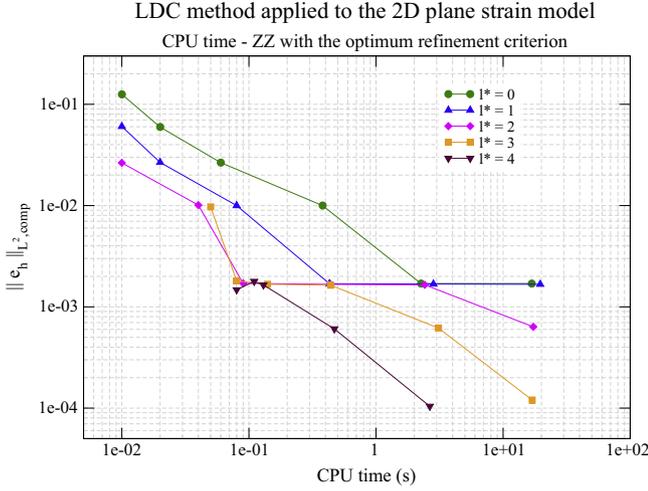


Fig. 21. Composite  $L^2$  error norm versus CPU time required for the simulation – The 2-D  $(r, \theta)$  test case.

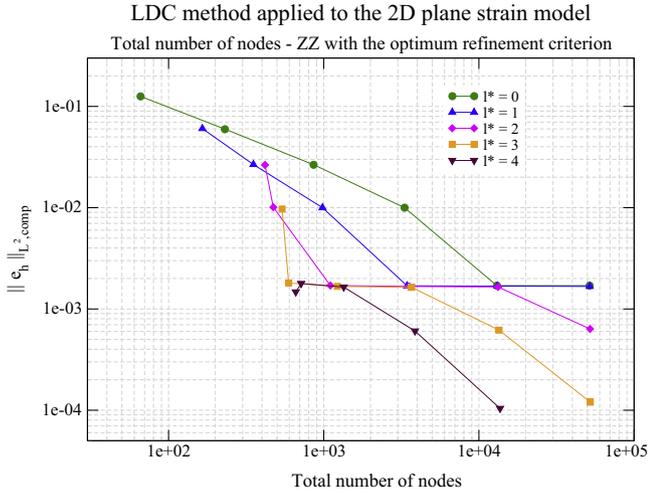


Fig. 22. Composite  $L^2$  error norm versus total number of nodes (the sum of each level) – The 2-D  $(r, \theta)$  test case.

refinement ratio of 2 as with one refinement with a refinement ratio of 4, even in the case of small errors. The convergence as  $O(d_{h_{fine}})$  was therefore conserved, whatever the refinement ratio tested. The results obtained in this study let think that our LDC method combined with the ZZ *a posteriori* error estimator seems to be generic (varying the refinement ratio, varying the number of sub-grids, etc.).

The second conclusion to be drawn here is that the use of a larger ratio reduces the number of sub-grids required but may induce the use of larger grids and thus a larger number of nodes than

Table 6 Comparison between refinement ratios of 2 and 4 – The 2-D  $(r, \theta)$  test case – Optimum refinement criterion.

$l^* \times r$	$2 \times 2$	$1 \times 4$	$4 \times 2$	$2 \times 4$
<i>Comparison between refinement ratios of 2 and 4 – <math>h_0 = h_i/2</math></i>				
$\ e_h\ _{L^2, comp}$	$1.01 \cdot 10^{-2}$	$1.02 \cdot 10^{-2}$	$1.78 \cdot 10^{-3}$	$1.79 \cdot 10^{-3}$
Sub-grid nodes	242	441	484	882
CPU time	0.04 s	0.08 s	0.11 s	0.13 s
<i>Comparison between refinement ratios of 2 and 4 – <math>h_0 = h_i/16</math></i>				
$\ e_h\ _{L^2, comp}$	$1.65 \cdot 10^{-3}$	$1.68 \cdot 10^{-3}$	$1.04 \cdot 10^{-4}$	$1.32 \cdot 10^{-4}$
Sub-grid nodes	242	441	638	1134
CPU time	2.42 s	3.62 s	2.65 s	3.70 s

necessary in some regions. In the present case, as the zones of interest were increasingly localised, there were around twice as many additional nodes with the ratio 4 than the ratio 2, for the same error level. The refinement ratio of 4 was therefore not found to be particularly attractive in terms of the total number of nodes in the present study. In addition, this ratio also involved slightly higher CPU time (up to 30% more) than a ratio of 2. However, as the CPU times were very short, this conclusion will have to be confirmed in further studies on a more representative test case.

### 5.3. Comparison with the h-refinement method

The aim of this section is to compare the LDC method with the standard approach applied until now to simulate the behaviour of cladding material exposed to irradiation. The standard meshes used for the PCI simulation were structured conforming mono-grid meshes with linear quadrangular elements (for modelling reasons) which were *priori*refined around the singularities (see figure 15. p.1621 in [76], for example). The obtained meshes were rather like those obtained with a global h-refinement method on quadrangular elements (see for example Fig. 23, left). The resulting meshes may contain degenerated elements.

#### 5.3.1. The two-dimensional plane strain test case

In view of the conclusions reached in Section 5.1, the LDC method seems to be efficient when a coarse initial mesh and a large number of local sub-grids are used. Further comparisons were made, taking an initial mesh size of  $h_0 = h_i/2$ , with 1 to 7 sub-grids and a refinement ratio  $r = 2$ . As we adopted an engineering point of view, the refinement criterion  $\alpha = 0.25$  was set for all accuracy levels. This value gives a good compromise between the number of elements to be refined and the error level reached (see the conclusions of Section 4.3). The meshes presented in Fig. 23 are examples of the meshes used in these comparisons. The composite  $L^2$  error norm is plotted versus the CPU time in Fig. 24 in the case of both approaches, and the total number of nodes are also indicated.

It can be concluded that the LDC solver is really efficient. With levels of errors of about  $1 \cdot 10^{-2}$ , the computational times are approximately twice as long in the case of the LDC method, but these times are nevertheless very short ( $\sim 0.05$  s). With smaller relative errors of less than  $1 \cdot 10^{-3}$ , the CPU times required to reach the resolution with the LDC mesh are less than 2 times shorter than with the h-refinement mesh.

In addition, our method does not require any preliminary studies in order to obtain a suitable refined mesh for dealing with the singularity problem. As the sub-grids are automatically generating using the ZZ *a posteriori* error estimator, only the coarsest mesh

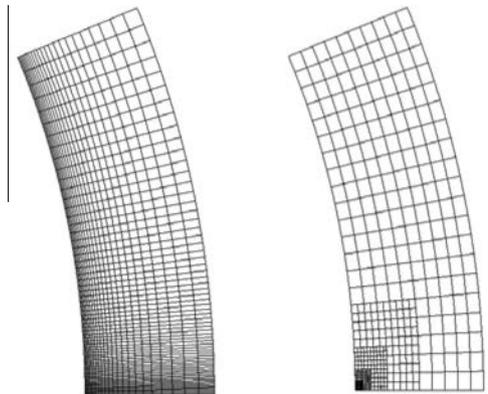
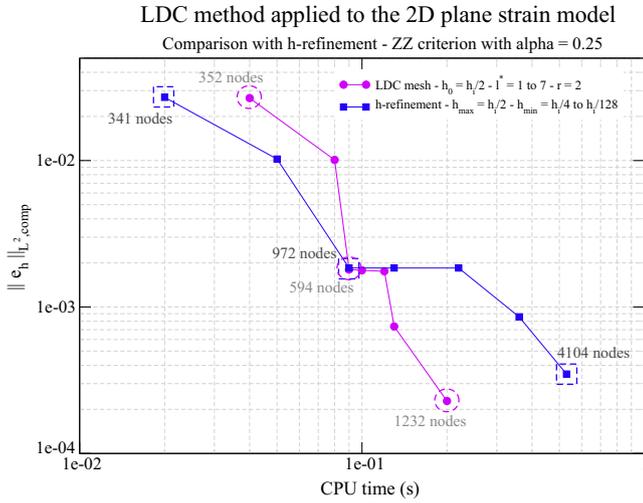


Fig. 23. Examples of meshes used in the comparative study: h-refined mesh (left) and LDC composite mesh with 4 sub-grids and a ratio of 2 (right).



**Fig. 24.** Relative composite  $L^2$  error norm depending on the CPU time and number of nodes – The 2-D plane strain test case – Comparisons between the LDC method and the global h-refinement method on linear quadrangular elements.

**Table 7**  
Comparisons between the LDC and h-refinement methods – The 3-D model.

Mesh	$\ e_h\ _{L^\infty, comp}$	CPU time	Number of nodes
LDC, $\alpha = 0.25$ $h_0 = h_i$ , $l^* = 2$	$2.61 \cdot 10^{-2}$	143 s	Total: 42813 max/level: 34121
h-refinement $h_{max} = h_i$	$2.35 \cdot 10^{-2}$	659 s	78819

size and a refinement criterion are necessary. The use of the LDC method therefore makes it possible to omit the preprocessing step (mesh determination), which can be of great importance from the engineering point of view.

The LDC method is also attractive from the memory space saving point of view. Even if extra information (boundary conditions on the levels, reversed stiffness matrices, right-hand side, etc.) has to be stored, each local grid is much smaller in terms of DoF than an equivalent overall refined grid. The total number of nodes in all the sub-grids is far smaller than the number of nodes in the currently used PCI locally refined mesh (there are half as many nodes with an error level of about  $1 \cdot 10^{-3}$  and 4 times fewer nodes with an error level of about  $1 \cdot 10^{-4}$  (see Fig. 24)).

To conclude, the performances of the LDC solver combined with the ZZ *a posteriori* error estimator were found to be highly efficient. Even on a simple test case which does not seem very suitable *a priori* for applying the LDC solver (2-D, linear elasticity, quick and accurate solving on a mono-grid stretched mesh), the LDC tool makes it possible to save CPU time and memory space at a given accuracy level.

### 5.3.2. The three-dimensional test case

Comparisons were also made in the case of a three-dimensional test between a similar *a priori* conforming refinement with quadrangular elements to that used in the 2-D case and the LDC solver (see Table 7).

These results are again most encouraging. At a maximum error level of about 2%, we obtained satisfactory results in terms of the CPU time and the memory space: the LDC solver gives savings amounting to 46% of the total number of nodes and 78% of the CPU time. It is also worth noting that with the LDC method, the number of nodes is the sum of all the nodes on all the grids. Hence, even the most costly level corresponds to 34121 nodes, which is only 43% of the nodes involved in the global h-refinement mesh.

As the number of nodes is the main restrictive factor in the inversion of the stiffness matrix, the LDC method will enable us to be subsequently limited by a finer local mesh step than for the standard approach. In particular, this means that this method overcomes one of the main limitations of industrial simulations. These performances could be further improved by using a domain decomposition method on patch elements [55].

Based on the performances obtained on the cases tested here, the LDC method can be said to be much more satisfactory than the global h-refinement method currently used in nuclear engineering simulations of the PCI. The same prescribed error level is obtained in shorter CPU times with less DoF, in 2-D as well as 3-D simulations. These results suggest that it would be worth performing further more detailed studies on the LDC solver.

## 6. Conclusions and prospects

The Local Defect Correction (LDC) multi-level method was tested here in a linear structural mechanics context and found to be of great potential interest. Some strategies are presented for obtaining the best possible performances with this method in engineering contexts. This efficient tool is based on the “black-box” solver concept and can be therefore easily applied with any existing industrial software. In addition, by construction, the LDC method can be combined with any *a posteriori* error estimator in order to generate local sub-grids automatically. In this study, the widely used classical Zienkiewicz and Zhu error estimator based on the super convergent patch recovery approach was used for this purpose. When applied to some non-commonplace test cases such as 3-D crossed-singularity problems in particular, the results obtained with the local multi-grid solver were as accurate as those obtained with the standard global h-refinement method, but the CPU time and memory space requirements were much lower.

These initial results suggest that the LDC solver may be also efficient with more complex structural mechanics behaviour, such as nonlinear path-dependent behaviour. This method has in fact already been successfully applied to “one-time-step” nonlinear behaviour (see [77]) and give promising results.

## Acknowledgements

This research was conducted in the framework of the PLEIADES project, which was supported financially by the CEA (Commissariat à l’Énergie Atomique et aux Énergies Alternatives), EDF (Électricité de France) and AREVA.

## References

- [1] Ghosh S, Manna S. R-adapted arbitrary Lagrangian–Eulerian finite-element method in metal-forming simulation. *J Mater Eng Perform* 1993;2(2):271–82.
- [2] Cao W, Huang W, Russell R. Comparison of two-dimensional r-adaptive finite element methods using various error indicators. *Math Comput Simul* 2001;56:127–43.
- [3] Strouboulis T, Haque K. Recent experiences with error estimation and adaptivity. 2. Error estimation for h-adaptive approximations on grids of triangles and quadrilaterals. *Comput Methods Appl Mech Eng* 1992;100(3):359–430.
- [4] Belytschko T, Tabbara M. H-adaptive finite-element methods for dynamic problems, with emphasis on localization. *Int J Numer Methods Eng* 1993;36(24):4245–65.
- [5] Fish J, Markolefas S. Adaptive global-local refinement strategy based on the interior error-estimates of the h-method. *Int J Numer Methods Eng* 1994;37(5):827–38.
- [6] Díez P, Huerta A. A unified approach to remeshing strategies for finite element h-adaptivity. *Comput Methods Appl Mech Eng* 1999;176:215–29.
- [7] Bessette G, Becker E, Taylor L, Littlefield D. Modeling of impact problems using an h-adaptive, explicit lagrangian finite element method in three dimensions. *Comput Methods Appl Mech Eng* 2003;192:1649–79.
- [8] Babuska I, Suri M. The optimal convergence rate of the p-version of the finite-element method. *SIAM J Numer Anal* 1987;24(4):750–76.

- [9] Barros F, Proenca S, de Barcellos C. Generalized finite element method in structural nonlinear analysis. A p-adaptive strategy. *Comput Mech* 2004;33(2):95–107.
- [10] Düster A, Rank E. The p-version of the finite element method compared to an adaptive h-version for the deformation theory of plasticity. *Comput Methods Appl Mech Eng* 2001;190:1925–35.
- [11] Fish J. The s-version of the finite-element method. *Comput Struct* 1992;43(3):539–47.
- [12] Yue Z, Robbins D. Adaptive superposition of finite element meshes in elastodynamic problems. *Int J Numer Methods Eng* 2005;63(11):1604–35.
- [13] Sun W, Zamani N. An adaptive h-r boundary element algorithm for the Laplace equation. *Int J Numer Methods Eng* 1992;33(3):537–52.
- [14] Askes H, Rodriguez-Ferran A. A combined rh-adaptive scheme based on domain subdivision. Formulation and linear examples. *Int J Numer Methods Eng* 2001;51(3):253–73.
- [15] Demkowicz L, Oden J, Rachowicz W, Hardy O. Toward a universal h-p adaptive finite-element strategy. 1. Constrained approximation and data structure. *Comput Methods Appl Mech Eng* 1989;77(1-2):79–112.
- [16] Oden J, Demkowicz L, Rachowicz W, Westermann T. Toward a universal h-p adaptive finite-element strategy. 2. A posteriori error estimation. *Comput Methods Appl Mech Eng* 1989;77(1-2):113–80.
- [17] Rachowicz W, Oden J, Demkowicz L. Toward a universal h-p adaptive finite-element strategy. 3. Design of h-p meshes. *Comput Methods Appl Mech Eng* 1989;77(1-2):181–212.
- [18] Valenciano J, Owens RG. An h-p adaptive spectral element method for Stokes flow. *Appl Numer Math* 2000;33:365–71.
- [19] Iskandarani M, Levin J, Choi B, Haidvogel D. Comparison of advection schemes for high-order h-p finite element and finite volume methods. *Ocean Model* 2005;10:233–52.
- [20] Haldenwang P, Pignol D. Dynamically adapted mesh refinement for combustion front tracking. *Comput Fluids* 2002;31:589–606.
- [21] Lebrun-Grandié D, Ragusa J, Turckin B. Adaptive multimesh hp-FEM for a coupled neutronics and nonlinear heat conduction problem. In: *International conference on mathematics and computational methods applied to nuclear science and engineering*. Rio de Janeiro, Brazil: American Nuclear Society; 2011. May 8–12.
- [22] Fournier D, Le Tellier R, Suteau C. Analysis of an a posteriori error estimator for the transport equation with  $S_N$  and discontinuous Galerkin discretizations. *Ann Nucl Energy* 2011;38:221–31.
- [23] Düster A, Niggel A, Rank E. Applying the hp-d version of the FEM to locally enhance dimensionally reduced models. *Comput Methods Appl Mech Eng* 2007;196:3524–33.
- [24] Solin P, Ceverny J, Dubcova L, Andrs D. Monolithic discretization of linear thermoelasticity problems via adaptative multimesh hp-FEM. *J Comput Appl Math* 2010;234:2350–7.
- [25] Ben Dhia H. Problèmes mécaniques multi-échelles: la méthode Arlequin. *CR Acad Sci Paris Ser II b* 1998;326:899–904 [in French].
- [26] Qiao H, Yang Q, Chen W, Zhang C. Implementation of the Arlequin method into ABAQUS: basic formulations and applications. *Adv Eng Softw* 2011;42(4):197–207.
- [27] Hackbusch W. Multi-grid methods and applications. Springer series in computational mathematics, vol. 4. Springer-Verlag; 1985.
- [28] Khadra K, Angot P, Caltagirone J, Morel P. Concept de zoom adaptatif en architecture multigrille locale; étude comparative des méthodes L.D.C., F.A.C. et F.L.C. *RAIRO Model Math Anal Numer* 1996;30(1):39–82 [in French].
- [29] Berger M, Oliger J. Adaptive mesh refinement for hyperbolic partial differential equations. *J Comput Phys* 1984;53(3):484–512.
- [30] Brandt A. Multi-level adaptive solutions to boundary-value problems. *Math Comput* 1977;31:333–90.
- [31] Hackbusch W. Local defect correction method and domain decomposition techniques. *Computing*, vol. 5. Springer-Verlag; 1984. p. 89–113.
- [32] Angot P, Caltagirone J, Khadra K. Une méthode adaptative de raffinement local: la correction de flux à l'interface. *CR Acad Sci Paris* 1992;315:739–45 [in French].
- [33] McCormick S. Fast adaptive composite grid (F.A.C.) methods: theory for the variational case. *Computing*, vol. 5. Springer-Verlag; 1984. p. 115–21.
- [34] Boussetta R, Coupez T, Fourment L. Adaptive remeshing based on a posteriori error estimation for forging simulation. *Comput Methods Appl Mech Eng* 2006;195(48-49):6626–45.
- [35] Biotteau E, Gravouil A, Lubrecht A, Combesure A. Multigrid solver with automatic mesh refinement for transient elastoplastic dynamic problems. *Int J Numer Methods Eng* 2010;84(8):947–71.
- [36] Bergallo M, Neumann C, Sonzogni V. Composite mesh concept based FEM error estimation and solution improvement. *Comput Methods Appl Mech Eng* 2000;188(4):755–74.
- [37] Stein E, de Borst R, Hughes T. *Encyclopedia of computational mechanics*, vol. 1. John Wiley & Sons, Ltd; 2004 [Chapter 4].
- [38] Babuska I, Rheinboldt W. A-posteriori error estimates for the finite element method. *Int J Numer Methods Eng* 1978;12:1597–615.
- [39] Ladevèze P, Leguillon D. Error estimate procedure in the finite-element method and applications. *SIAM J Numer Anal* 1983;20:485–509.
- [40] Zienkiewicz O, Zhu J. A simple error estimator and adaptive procedure for practical engineering analysis. *Int J Numer Methods Eng* 1987;24:337–57.
- [41] Gallimard L, Ladevèze P, Pelle J. An enhanced error estimator on the constitutive relation for plasticity problems. *Comput Struct* 2000;78:801–10.
- [42] Gratsch T, Bathe K. Review – a posteriori error estimation techniques in practical finite element analysis. *Comput Struct* 2005;83:235–65.
- [43] Belliard M, Grandotto M. Local zoom computation of two-phase flows in steam generators using a local defect correction method. *Numer Heat Transfer Part A Appl* 2003;43(2):111–35.
- [44] Angot P, Laugier M. The FIC method of conservative connection between nested subdomains for an ocean circulation model. *CR Acad Sci Paris Ser II Part 1* 1994;319(9):993–1000.
- [45] Coré X, Angot P, Latché J-C. A multilevel local mesh refinement projection method for low Mach number flows. *Math Comput Simul* 2003;61(3-6):477–88.
- [46] Kramer W, Clercx H, Mattheij R, Minero R. A finite volume local defect correction method for solving the transport equation. *Comput Fluids* 2009;38(3):533–43.
- [47] Watremetz B, Baietto-Dubourg M, Lubrecht A. 2D thermo-mechanical contact simulations in a functionally graded material: a multigrid-based approach. *Tribol Int* 2007;40:754–62.
- [48] Ferket P, Reusken A. Further analysis of the local defect correction method. *Computing* 1996;56(2):117–39.
- [49] Anthonissen M, Mattheij R, Boonkamp J. Convergence analysis of the local defect correction method for diffusion equations. *Numer Math* 2003;95(3):401–25.
- [50] Carre G, Dervieux A. On the application of FMG to variational approximation of flow problems. *Int J Comput Fluid Dyn* 1999;12(2):99–117.
- [51] Michel B, Sercombe J, Thouvenin G, Chatelet R. 3D fuel cracking modelling in pellet cladding mechanical interaction. *Eng Fract Mech* 2008;75:3581–98.
- [52] Nonon C, Lansart S, Struzik C, Plancq D, Martin S, Decroix G, et al. Differential PCI behaviour of PWR fuel rods under transient conditions. In: *International topical meeting on LWR fuel performance*. Orlando, FL: American Nuclear Society; 2004. September 19–22.
- [53] Roberts G. The concentration of stress in cladding produced by the expansion of cracked fuel pellets. *Nucl Eng Des* 1978;47:257–66.
- [54] Minero R, Anthonissen M, Mattheij R. A local defect correction technique for time-dependent problems. *Numer Methods Partial Differ Equ* 2006;22:128–44.
- [55] Anthonissen M, Bennet B, Smooke M. An adaptive multilevel local defect correction technique with application to combustion. *Combust Theory Model* 2005;9:273–99.
- [56] Anthonissen M, van 't Hof B, Reusken A. A finite volume scheme for solving elliptic boundary value problems on composite grids. *Computing* 1998;61:285–305.
- [57] Wappler JU. Die lokale defektkorrekturmethode zur adaptiven diskretisierung elliptischer differentialgleichungen mit finiten elementen. Ph.D. thesis, Christian-Albrechts-Universität, Kiel; 1999 [in German].
- [58] Lemke M, Quinlan D. Fast adaptive composite grid methods on distributed parallel architectures. *Commun Appl Numer Methods* 1992;8(9):609–19.
- [59] Ritter D, Sturmer M, Rude U. A fast-adaptive composite grid algorithm for solving the free-space Poisson problem on the cell broadband engine. *Numer Linear Algebra Appl* 2010;17(2-3):291–305.
- [60] Ramière I, Angot P, Belliard M. A general fictitious domain method with immersed jumps and multilevel nested structured meshes. *J Comput Phys* 2007;225:1347–87.
- [61] Nefedov V, Mattheij R. Local defect correction with different grid types. *Numer Methods Partial Differ Equ* 2002;18(4):468–545.
- [62] Babuska I, Strouboulis T, Marthar A, Upadhyay C. Pollution-error in the h-version of the finite-element method and the local quality of a-posteriori error estimators. *Finite Elem Anal Des* 1994;17:273–321.
- [63] Ciarlet PG. The finite element method for elliptic problems. In: *Lions J, Papanicolaou G, Rockafellar RT, editors. Studies in mathematics and its applications*, vol. 4. North-Holland Publishing Company; 1978.
- [64] CAST3M, <www.cast3m.cea.fr>.
- [65] Raviart P-A, Thomas J-M. *Introduction à l'analyse numérique des équations aux dérivées partielles*. In: *Ciarlet PG, Lions JL, editors. Dunod Edition; 1998 [in French]*.
- [66] Ramière I. Convergence analysis of the  $Q_1$ -finite element method for elliptic problems with non-boundary-fitted meshes. *Int J Numer Methods Eng* 2008;75(9):1007–52.
- [67] Kunert G, Nicaise S. Zienkiewicz-Zhu error estimators on anisotropic tetrahedral and triangular finite element meshes. *ESAIM Math Model Numer Anal Model Math Anal Numer* 2003;37(6):1013–43.
- [68] Wriggers P, Scherf O. Different a posteriori error estimators and indicators for contact problems. *Math Comput Model* 1998;28(4-8):437–47.
- [69] Bouillard P, Allard J, Warzee G. Superconvergent patch recovery technique for the finite element method in acoustics. *Commun Numer Methods Eng* 1996;12(9):581–94.
- [70] Murthy K, Mukhopadhyay M. Adaptive finite element analysis of mixed-mode crack problems with automatic mesh generator. *Int J Numer Methods Eng* 2000;49(8):1087–100.
- [71] Zienkiewicz O, Zhu J. The superconvergent patch recovery and a posteriori error estimation. Part I: The recovery technique. *Int J Numer Methods Eng* 1992;33:1331–64.
- [72] Zienkiewicz O, Zhu J. The superconvergent patch recovery and a posteriori error estimation. Part II: Error estimates and adaptivity. *Int J Numer Methods Eng* 1992;33:1365–82.

- [73] Desroches X. Estimateur d'erreur de Zhu–Zienkiewicz en élasticité 2D, manuel de référence Code\_Aster; 2009 [in French].
- [74] Biotteau E, Gravouil A, Lubrecht A, Combescure A. Three dimensional automatic refinement method for transient small strain elastoplastic finite element computations. *Comput Mech* 2012;49(1):123–36.
- [75] Ramière I, Angot P, Belliard M. A fictitious domain approach with spread interface for elliptic problems with general boundary conditions. *Comput Methods Appl Mech Eng* 2007;196:766–81.
- [76] Michel B, Sercombe J, Thouvenin G. A new phenomenological criterion for pellet-cladding interaction rupture. *Nucl Eng Des* 2008;238:1612–28.
- [77] Barbié L, Ramière I, Lebon F. A multilevel technique based on nested local meshes for nonlinear mechanics. In: Topping B, editor. *Proceedings of the eighth international conference on engineering computational technology*. Stirlingshire, UK: Civil-Comp Press; 2012. <http://dx.doi.org/10.4203/ccp.100.88> [paper 88, September 4–7, Dubrovnik, Croatia].