A weakly-intrusive multi-scale substitution method in explicit dynamics.

Omar Bettinotti

To cite this version:

Omar Bettinotti. A weakly-intrusive multi-scale substitution method in explicit dynamics.. Structural mechanics [physics.class-ph]. École normale supérieure de Cachan - ENS Cachan, 2014. English. NNT : . tel-01073520

HAL Id: tel-01073520
https://theses.hal.science/tel-01073520

Submitted on 10 Oct 2014

ENSC-2013/14

THÈSE DE DOCTORAT
DE L'ÉCOLE NORMALE SUPÉRIEURE DE CACHAN

Présentée par

Omar Bettinotti

pour obtenir le grade de

DOCTEUR DE L'ÉCOLE NORMALE SUPÉRIEURE DE CACHAN

Domaine

MÉCANIQUE - GÉNIE MÉCANIQUE - GÉNIE CIVIL

Sujet de la thèse

A weakly-intrusive multi-scale Substitution method in explicit dynamics

Soutenue à Cachan le 17 septembre 2014 devant le jury composé de :

Anthony Gravouil	Professeur, INSA de Lyon	Rapporteur
Peter Wriggers	Professeur, Leibniz Universität Hannover	Rapporteur
Olivier Allix	Professeur, ENS de Cachan	Directeur de thèse
Benoît Malherbe	Ingénieur, Airbus Operations	Examinateur
Victor Oancea	Ingénieur, Dassault Systèmes Simulia	Examinateur
Umberto Perego	Professeur, Politecnico di Milano	Examinateur
Gérard Rio	Professeur, Université de Bretagne-Sud	Examinateur

LMT-Cachan
ENS Cachan / CNRS / UPMC / PRES UniverSud Paris
61 avenue du Président Wilson, F-94235 Cachan cedex, France

To Benedetta and my parents

A day without laughter
is a day wasted.

— Charlie Chaplin

At the end of this wonderful experience, I would like to express all my gratitude to my supervisors Prof. Olivier Allix and Ing. Benoît Malherbe for all the trust, for all the discussions and simply for all the time that we spent together in this really compelling project.

A special thank goes to Dr. Victor Oancea, for the constant support and for all the help on the other side of the ocean.

I would like to thank Prof. Umberto Perego, once again, for his important contribution to the achievement of the research targets.

I would like to thank all my friends and colleagues at LMT-Cachan, coming from all over the world, who made this experience unique.

I would like to thank the entire Vulnerability Group of Airbus in Toulouse and all the colleagues in Dassault Systèmes Simulia in Providence, because in both places I have always felt at home.

A great thank goes to my parents, who have always encouraged me every time and everywhere.

The greatest thank goes to Benedetta, together day by day, simply letting everything be possible. Thank you!

Résumé

Dans le contexte de la simulation numérique des structures avec caractéristiques fines (micro-fractures, petits défauts ou phénomènes à une échelle fine), le besoin d'appliquer modèles fins est toujours en antithèse avec les temps de calcul souhaitables pour la pratique de l'ingénieur. Pour réduire ces temps de calcul, les techniques multi-échelles permettent de coupler un modèle fin appliqué seulement où nécessaire à un modèle grossier appliqué au reste de la structure. Techniques de couplage multi-échelles sont aujourd'hui disponibles aussi en dynamique transitoire. Dans ce contexte, l'utilisation d'un schéma d'intégration en temps explicite requiert l'application d'une discrétisation en temps en function de taille de maillage et modèle de matériau. Le couplage multi-échelles doit donc être appliqué en espace et en temps.

Parmi les méthodes les plus avancées, la méthode de décomposition de domaines permet de découpler le domaine du problème en plusieurs sous-domaines et d'appliquer différents pas de temps sur les différents sous-domaines. En revanche, en présence de phénomènes évolutifs, typiques de la dynamique transitoire comme dans le cas de délaminage pour les composites stratifiés sous impact, une adaptabilité dynamique est nécessaire et exige une stratégie de re-maillage et une re-définition des sous-domaines. Une telle stratégie est considérée comme intrusive pour la mise en œuvre dans logiciels commerciaux pour lanalyse à éléments finis.

Dans ce travail, les bases d'une approche faiblement intrusive, appelée méthode de substitution, sont proposées. La méthode s'appuie sur une formulation globale-locale, conçue de telle manière que une analyse globale grossière est appliquée à la totalité du domaine et corrigée par une analyse locale fine définie seulement où nécessaire. De cette manière, la stratégie d'adaptation peut être entraînée par l'activation ou la désactivation de taches locales complètement superposées au domaine global. Le prix de calcul à payer est que la méthode est “localement” itérative. La vérification de la méthode de substitution est conduite par rapport à la méthode de décomposition de domaines, en parallel à un étude de la stabilité du couplage multi-pas de temps. L'amélioration de l'efficacité du processus itératif est ensuite discutée. L'efficacité et la robustesse de la nouvelle procédure sont démontrées à travers des exemples de complexité croissante. De cette façon, la possibilité de prendre en compte des défauts et des lois de comportements non-linéaires avec endommagement, ainsi que l'adaptation évolutive de l'échelle locale pour le cas du délaminage dans les matériaux composites, est également décrite. Enfin, la mise en œuvre de la méthode au sein du code commercial Abaqus/Explicit est abordée.

Contents

List of Figures

List of Tables

Introduction

In the design and certification process of the aircrafts, it is appropriate to carry out qualitative and then quantitative vulnerability analyses. This approach ensures the aircraft configuration to be optimized in order to reduce vulnerability and thus maximize survivability. Vulnerability analyses include crash, emergency landing, explosion or impact. The aircraft structures subjected to these test cases are various and complex and engineers are used to deal with several challenges of structural dynamics.

The increasing use of composite materials in the vulnerable components of the aircraft gives rise to additional situations to deal with. Indeed, the same safety levels of metallic structures must be attained in the composite ones. In particular, composite laminated materials are prone to extensive damage phenomena, such as delamination or matrix cracking, when submitted to impact loads.

The necessity to perform virtual testing in transient dynamics, e.g. for impact problems on composites for predicting the extension of damage in the structure, becomes essential for the engineering workflows. To be reliable, such virtual testing should preferably employ a fine modeling scheme, the so-called *meso*-scale model, with solid discretization of each ply and use of cohesive interfaces. However, the computational cost associated with such modeling scheme for large structures would be prohibitively high in engineering practice, as the precise study of damage and failure response requires the consideration of phenomena encompassing multiple scales both in space and time.

Nowadays, large composite structures are modeled for impact simulations with multi-layered shell elements, the so-called *macro*-scale model, which plies' stiffness is governed by a damage law. Once almost all the plies are completely damaged, the shell element is eliminated, in order to reproduce structural cracks or perforations with acceptable accuracy. Such kind of *macro*-scale model is not capable of simulating the delamination mechanism and the associated inter-ply damage.

A so-called *pseudo-meso*-scale model is then employed and calibrated when the test case experience involves diffused delamination. In such model, described for instance in Figure 1, some multi-layered shell elements are stacked and some finer cohesive interfaces are inserted between them. Important parameters of the calibration are the number and the position of the cohesive interfaces. The number of degrees of freedom in Figure 1 is so reduced with respect to the detailed *meso*-scale model, but not so much to be generally applied to large structures.

Nonetheless, as stated in the model validation in [Heimbs et al., 2014] for high-velocity impacts on a composite flat panel, a detailed *meso*-scale model is preferable

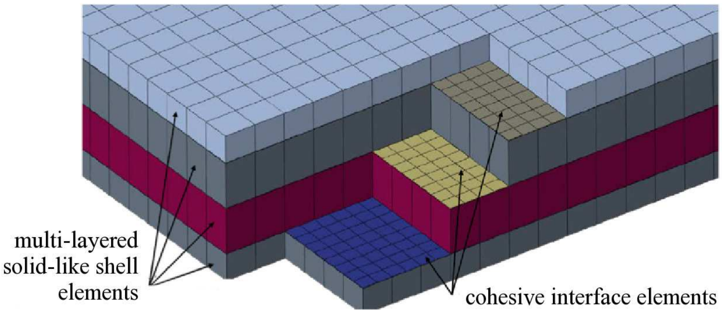
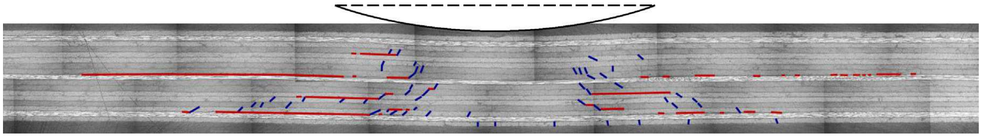
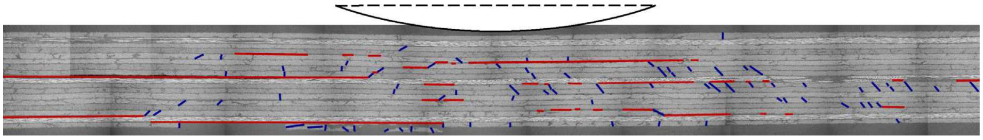


Figure 1: Pseudo-meso-scale model used in simple industrial test cases.

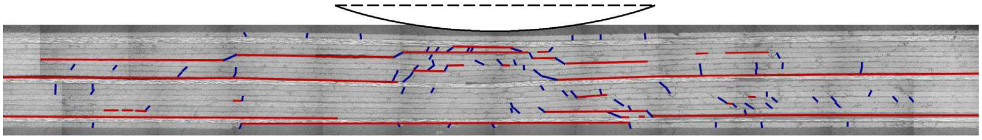
for predicting the delamination with acceptable accuracy and robustness. From the above mentioned paper, Figure 2 shows the experimental results obtained impacting a stratified composite with a stacking sequence of 17 plies for a nominal thickness of 2.125 *mm* with a glass projectile of diameter $\phi = 25$ *mm* and mass $m = 21$ *g* at different velocities. Micrographs are taken at representative positions directly under the impact point in order to analyze the number and extension of delamination interfaces (red lines) and matrix cracks (blue lines). It is shown that the damage configuration is different for each impact velocity and for different pre-loads: only a detailed *meso*-scale model with 16 cohesive interfaces is capable to represent the correct delamination process for all the test cases.



(a) impact velocity: 54 m/s



(b) impact velocity: 70 m/s



(c) impact velocity: 84 m/s

Figure 2: Composite panel impacted by a glass projectile [Heimbs et al., 2014].

The necessity to develop a multi-scale technique bridging the *macro*- and *meso*-scales then represents a new challenge for the virtual testing of laminated composites in transient dynamics.

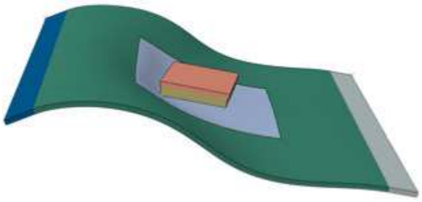
The aim of this work is to develop an operational multi-scale coupling methodology, in order to use a *meso*-scale model for simulating the delamination propagation, but limiting its zone of action only where needed, using a *macro*-scale model in the rest of the structure. Based on the size of the structure, such strategy should be able to optimize the use of the detailed model and to decrease the computational costs.

On the other hand, industrial groups tend to perform analogous sets of numerical simulations with a unique generic Finite Element software package, exploiting the existing variety of engineering tools. In this scenario, the development of a multi-scale methodology has to deal with a new issue: the intrusivity of the insertion of a given method within an existing commercial software. For this purpose, a global-local iterative solver was introduced by [Whitcomb, 1991] to couple a coarse and homogeneous global model applied to the overall structure, with a detailed local model, e.g. with the presence of defects. Inspired to this approach, a non-intrusive global-local coupling technique is described in [Gendre et al., 2009] and [Gendre et al., 2011] for a nonlinear static application: Abaqus/Standard, utilized as global linear elastic analysis, has been locally coupled with a research code, used to reproduce localized nonlinear behaviors. Another example of non-intrusive coupling is described in [Plews et al., 2012], in which a thermal shock computation is coupled with a Finite Element analysis platform. Other applications are devoted to fracture problems solved with local patches as in [Kerfriden et al., 2012], [Gupta et al., 2012] or [Passieux et al., 2013]. At last, a non-intrusive application to shell-to-solid coupling and to the treatement of defects in the refined solid model is presented in [Guguin et al., 2014].

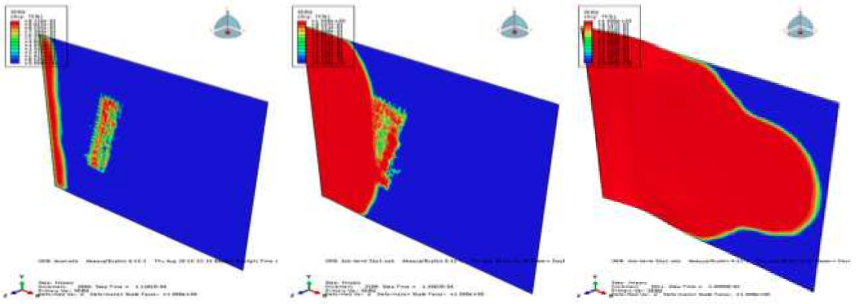
The need to couple multiple scales is strongly connected to the need of a methodology suitable for the engineering workflows, meaning that the solving procedure must be compatible with the existing features of the commercial software. Furthermore, the majority of the technologies enumerated above are applied to static or quasi-static problems. The extension of such methodologies to transient dynamics is not straightforward, because the issues and algorithmic strategies are highly different.

Impact analyses and, more generally, transient dynamic problems are usually performed with explicit solution schemes (e.g. Abaqus/Explicit). In this field, today's technologies allow the engineers to couple *macro*- and *meso*-scale models, e.g. employing non-uniform meshes and Subcycling techniques as in [Belytschko et al., 1979] and [Daniel, 2003] or Domain Decomposition-based coupling methods as in [Gravouil and Combescure, 2001] and following works. These techniques are based on the *a priori* subdivision and calibration of the models topology, as for instance in [Chantrait et al., 2014] for an implicit-explicit *macro-meso*-scale coupling in low-energy impact applications. In highly transient dynamics, the *a priori* subdomains partitioning hinders the automatization of the design process, because it forces to a pre-definition of the zones covered by the *meso*-scale model, i.e. of all the zones covered by the delamination process during the analysis.

Let's consider, for instance, Figure 3: it shows a tire debris impact analysis. An initially deformed panel is impacted by a tire debris. The composite panel is modeled with a *pseudo-meso*-scale model composed of 2 plies and 1 cohesive interface. The boundary conditions are imposed along the blue and gray extremities of the panel and the delamination process is monitored in the gray impacted zone (red color denotes a complete damage, blue color denotes absence of damage). Numerical results show a quick movement of the delamination process zone (intermediate colors between red and blue), traversing almost the whole monitored patch. Nevertheless, the delamination process zone is always localized to a very limited portion of the domain.



(a) composite and tire debris models



(b) 64% of the analysis (c) 70% of the analysis (d) 100% of the analysis

Figure 3: Initially deformed composite panel impacted by a tire debris and delamination (at the bottom) over the gray zone of the structure (at the top).

Thus, an evolutive delamination process requires the use of re-meshing strategies in order to adapt the models' topology to the process zone position and limit the size of the *meso*-scale model domain. Nevertheless, a mesh generation during a dynamic analysis, when possible and stable, could be considered to be an inefficient and intrusive implementation. A new methodology allowing the model adaptation is so desirable. A submodeling-like two-way coupling technique seems to be more suitable for the scope, because it allows one to apply the *macro*-scale model to the overall structure (as global analysis) and to overlap the *meso*-scale model over the zones of interest (as local analysis) via specific pre-defined activation criteria. The dynamic conditions that connect the spatial scale to the temporal one give rise to additional challenges.

Outline

This work describes the development and implementation of the proposed Finite Element weakly-intrusive coupling method for the simulation of impact tests on composite structures. Firstly, a bibliographic research is presented, focused on advantages and issues of existing technologies, from the explicit dynamics properties to the recent coupling methodologies. Some of the most recent coupling technologies have been deeply investigated with the implementation in Matlab prototypes, in order to understand potentialities and issues and also to collect reference results. Then, a new methodology is introduced with studies of accuracy, stability and efficiency through Matlab prototypes and verifications in comparison with the reference coupling methods. Finally, the implementation of the methodology inside the existing commercial software Abaqus is described and the related features are discussed.

Besides the present **Introduction**, the manuscript is organized according to the following scheme:

- **Part 1** presents the context and the existing technologies:
    - **Chapter 1** introduces the advanced methodologies in the field of transient dynamics. Starting from a generic initial-boundary value problem formulation, the main time integration techniques are presented, specifying the explicit time integration scheme, its properties and related resolution strategy. The main coupling schemes in the field of explicit dynamics are then enumerated and formulated, focusing on advantages and disadvantages. The existing capabilities inside the commercial software Abaqus are summarized. Motivations of the further developments and implementations are discussed.
    - **Chapter 2** focuses the attention on the evolution of the non-overlapping Domain Decomposition method for transient dynamics. Details of the algorithms are given and related results are preliminarily analyzed and kept as reference for the following developments. The advantages and disadvantages of the algorithm [Gravouil and Combescure, 2001] put in evidence and compared with the energy-preserving algorithm proposed in [Prakash and Hjelmstad, 2004] and reformulated in [Mahjoubi et al., 2009].
- **Part 2** presents the development and implementation of the new substitution-based methodology, built upon the considerations related to the bibliographic study.
    - **Chapter 3** introduces the formulation of the substitution-based coupling strategy. The methodology is presented and verified in comparison with the energy-preserving Domain Decomposition algorithm. A convergence study refining the time steps is illustrated and the stability of the methodology is analyzed applying the energy method described in [Hughes and Liu, 1978b]. The remarks given in [Prakash and Hjelmstad, 2004] are analogously applied in order to obtain an energy-preserving substitution-based coupling method.
    - **Chapter 4** is devoted to the improvement of the iterative scheme as a result of time integration properties, in order to meet the efficiency targets required by the industrial application. The possibility to treat defects and nonlinear

damage behaviors is verified and an evolution-adaptive application is also described.

At the end, **Conclusions and prospects** close the manuscript with an overall summary of the work and a list of potential future developments.

Part I

State of the art and focus on non-overlapping Domain Decomposition methods

Chapter 1

Transient dynamics and existing Finite Element methodologies

This Chapter introduces the Finite Element methodologies in the field of transient dynamics. An explicit time integration scheme is considered, exploring its properties and resolution strategies. The advantages and disadvantages of the multi-scale coupling schemes are evinced. The existing coupling techniques inside the commercial software Abaqus are presented. Motivations of the further developments and implementations are discussed.

1 Introduction

In this chapter, starting from the strong form of the initial-boundary value problems for dynamic analysis, the most common time integration techniques are presented. Among the Newmark's time integration schemes, introduced in [Newmark, 1959], the Central Difference scheme is highlighted as the preferable technique for the industrial goal of the present work, because leading to characteristics particularly suitable to transient dynamics. Alternative interesting schemes are also underlined, even if not considered in further developments.

Advanced multi-scale coupling techniques are then described. Such techniques aim at partitioning the overall domain in different scale models and discretizations, in function of the different refinement needs. Domain Decomposition methods are introduced with the explicit dynamic formulation and classified as non-overlapping and overlapping techniques.

Last, but not least, the existing coupling techniques inside the commercial software Abaqus are evaluated, in order to recognize eventual missing bricks for the industrial goal achievement.

2 Initial-boundary value problem

For the sake of simplicity, a generic initial-boundary value problem for dynamic analysis of structures is considered here, which is a case sufficient to introduce the main concepts that will be used in the thesis. This is not the case when considering contact or impact problems, in which formulation has to be adopted according for instance to [Wriggers and Simo, 1985] for nonlinear contact, to [Wriggers et al., 1990] or [Oancea and Laursen, 1997] for frictional effects, to [Wriggers, 2002] for a wide description of computational procedures.

The initial-boundary value problem for dynamics of structures, defining the spatial domain $\Omega \subset \mathbb{R}^{\mathcal{N}}$, in the strong form reads:

$$\begin{aligned}
&\rho\ddot{\mathbf{u}} = \mathrm{div}(\boldsymbol{\sigma}) + \mathbf{f}_{\mathrm{body}} && \text{in } \Omega \times [t_{\mathrm{in}}, t_{\mathrm{end}}] \\
&\mathbf{u} = \bar{\mathbf{u}} && \text{over } \partial\Omega_u \times [t_{\mathrm{in}}, t_{\mathrm{end}}] \\
&\boldsymbol{\sigma} \cdot \mathbf{n} = \mathbf{f}_{\mathrm{surf}} && \text{over } \partial\Omega_f \times [t_{\mathrm{in}}, t_{\mathrm{end}}] \\
&\{\mathbf{u}, \dot{\mathbf{u}}\} = \{\mathbf{u}_0, \mathbf{v}_0\} && \text{in } \Omega|_{t_{\mathrm{in}}}
\end{aligned} \tag{1.1}$$

where ρ is the density of the material, $\boldsymbol{\sigma}$ is the stress tensor, $\mathbf{f}_{\mathrm{body}}$ indicates the body forces, $\mathbf{f}_{\mathrm{surf}}$ is the impact surface load, symbols $\cdot$ and $\times$ indicate scalar and vector products, respectively, $\mathbf{n}$ is the unit normal to the boundary $\partial\Omega_f$, $\mathbf{u}_0$ and $\mathbf{v}_0$ are the initial displacement and velocity, respectively. The times t_{in} and t_{end} denote the initial and the final instants, respectively. The stress tensor $\boldsymbol{\sigma}$ is related to the strain tensor $\varepsilon(\mathbf{u})$ by a material constitutive law.

The first equation of System (1.1) represents the momentum balance in terms of the displacements $\mathbf{u}(\mathbf{x}, t)$, $\mathbf{x}$ being the position in the domain Ω and t being the time varying in $[t_{\text{in}}, t_{\text{end}}]$. The second and the third ones denote essential and natural boundary conditions, respectively. The fourth one denotes initial conditions. The prescribed boundary displacements are applied to the boundary $\partial\Omega_u$ and the prescribed tractions are applied to the boundary $\partial\Omega_f$. The union between $\partial\Omega_u$ and $\partial\Omega_f$ constitutes the entire boundary of the problem $\partial\Omega$, so that $\partial\Omega_f \cup \partial\Omega_u \equiv \partial\Omega$ and $\partial\Omega_f \cap \partial\Omega_u \equiv \emptyset$.

The problem introduced in System (1.1) is rewritten in the weak formulation as find $\mathbf{u}(\mathbf{x},t)$ such that:

$$\begin{gathered}\int_\Omega (\rho \ddot{\mathbf{u}} \cdot \delta\mathbf{u} + \boldsymbol{\sigma}(\mathbf{u}) : \varepsilon(\delta\mathbf{u}))\ d\Omega = \int_\Omega \mathbf{f}_{\text{body}} \cdot \delta\mathbf{u}\ d\Omega + \int_{\partial\Omega_f} \mathbf{f}_{\text{surf}} \cdot \delta\mathbf{u}\ d\Gamma \\ \forall \delta\mathbf{u} = \mathbf{0} \text{ over } \partial\Omega_u \times [t_{\text{in}}, t_{\text{end}}]\end{gathered} \tag{1.2}$$

where the symbol : indicates a scalar product between two tensors of order 2.

A displacement-based Finite Element method (see [Hughes, 1987], [Bathe, 1996] or [Zienkiewicz and Taylor, 2000] for basics) is employed in order to numerically integrate in space the weak formulation of the problem in Equation (1.2) via the mesh Ω^h, so that:

$$\mathbf{u}(\mathbf{x},t) = \boldsymbol{\phi}(\mathbf{x}) \cdot \mathbf{U}(t) \tag{1.3}$$

where $\boldsymbol{\phi}(\mathbf{x})$ denotes the Finite Elements space discretization and $\mathbf{U}$ indicates the displacements parameters varying in time.

The following system of Ordinary Differential Equations written in the matrix notation is obtained:

$$\begin{array}{ll} \mathbb{M}\ddot{\mathbf{U}} + \mathbf{F}^{\text{int}}(\mathbf{U}) = \mathbf{F}^{\text{ext}} & \text{in } \Omega^h \times [t_{\text{in}}, t_{\text{end}}] \\ \mathbf{U} = \overline{\mathbf{U}} & \text{over } \partial\Omega_u \times [t_{\text{in}}, t_{\text{end}}] \\ \{\mathbf{U}, \dot{\mathbf{U}}\} = \{\mathbf{U}_0, \mathbf{V}_0\} & \text{in } \Omega^h|_{t_{\text{in}}} \end{array} \tag{1.4}$$

where $\mathbb{M}$ is the consistent mass matrix, $\mathbf{F}^{\text{ext}}$ and $\mathbf{F}^{\text{int}}$ are the external and internal force vectors, respectively, $\mathbf{U}_0$ and $\mathbf{V}_0$ are the initial nodal displacements and velocities, respectively.

3 Time integration

System (1.4) is then numerically integrated in time with a discretization of the time domain $[t_{\text{in}}, t_{\text{end}}]$. For the sake of simplicity, a constant time stepping is used, so that $[t_{\text{in}}, t_{\text{end}}]$ is partitioned in N time steps Δt of the same size, with $t_{\text{end}} - t_{\text{in}} = N\Delta t$.

For the given mesh Ω^h at a generic time $t^n \in [t_{\text{in}}, t_{\text{end}}]$, nodal displacements, velocities and accelerations are respectively defined as:

$${}^n\mathbf{U} = \mathbf{U}(t^n); \quad {}^n\mathbf{V} = \dot{\mathbf{U}}(t^n); \quad {}^n\mathbf{A} = \ddot{\mathbf{U}}(t^n). \tag{1.5}$$

The nodal velocities at the time instant t^{n+1} are obtained adding the integral of the second temporal derivative of the nodal displacements over the time interval $[t^n, t^{n+1}]$ to the nodal velocities at the time instant t^n, so that:

$${}^{n+1}\mathbf{V} = {}^n\mathbf{V} + \int_{t^n}^{t^{n+1}} \ddot{\mathbf{U}}\ dt. \tag{1.6}$$

By analogy, the nodal displacements at the time t^{n+1} are obtained with the integral of the first temporal derivative of the nodal displacements over the time interval $[t^n, t^{n+1}]$. This last is then integrated by parts in order to obtain an integral of the second temporal derivative of the nodal displacements, so that:

$$^{n+1}\mathbf{U} = {}^{n}\mathbf{U} + {}^{n}\mathbf{V}\Delta t + \int_{t^n}^{t^{n+1}} (t^{n+1} - t)\ddot{\mathbf{U}}\, dt \tag{1.7}$$

with $\Delta t = t^{n+1} - t^n$.

Following Newmark's integration scheme introduced in [Newmark, 1959] on the basis of the Taylor's expansion, the two time integrations of the second derivative of displacements are defined with the use of two parameters $\gamma \in [0,1]$ and $\beta \in [0,1/2]$ as:

$$\int_{t^n}^{t^{n+1}} \ddot{\mathbf{U}}\, dt = {}^{n}\mathbf{A}(1-\gamma)\Delta t + {}^{n+1}\mathbf{A}\gamma\Delta t + \mathbf{U}^{(iii)}\left(\frac{1}{2} - \gamma\right)\Delta t^2 + O\left(\mathbf{U}^{(iv)}\Delta t^3\right) \tag{1.8}$$

where superscripts *(iii)* and *(iv)* denote the third and fourth derivatives in time, respectively, and:

$$\begin{aligned}\int_{t^n}^{t^{n+1}} (t^{n+1} - t)\ddot{\mathbf{U}}\, dt =& {}^{n}\mathbf{A}\left(\frac{1}{2} - \beta\right)\Delta t^2 + {}^{n+1}\mathbf{A}\beta\Delta t^2 + \\ &+ \mathbf{U}^{(iii)}\left(\frac{1}{6} - \beta\right)\Delta t^3 + O\left(\mathbf{U}^{(iv)}\Delta t^4\right).\end{aligned} \tag{1.9}$$

Truncating Equation (1.8) to an error $O\left(\mathbf{U}^{(iii)}\Delta t^2\right)$ for the approximation of Equation (1.6), the nodal velocities are so updated with:

$$^{n+1}\mathbf{V} = {}^{n}\mathbf{V} + {}^{n}\mathbf{A}(1-\gamma)\Delta t + {}^{n+1}\mathbf{A}\gamma\Delta t \tag{1.10}$$

and truncating Equation (1.9) to an error $O\left(\mathbf{U}^{(iii)}\Delta t^3\right)$ for the approximation of Equation (1.7), the nodal displacements are updated with:

$$^{n+1}\mathbf{U} = {}^{n}\mathbf{U} + {}^{n}\mathbf{V}\Delta t + {}^{n}\mathbf{A}\left(\frac{1}{2} - \beta\right)\Delta t^2 + {}^{n+1}\mathbf{A}\beta\Delta t^2. \tag{1.11}$$

Note that, choosing $\gamma = 1/2$, the third term on the right-hand side of Equation (1.8) vanishes and the error of the velocities' approximation remains $O\left(\mathbf{U}^{(iv)}\Delta t^3\right)$.

Remark 1 *In this scenario, the Newmark's integration schemes are based on the approximation of the displacement and velocity fields in function of the unknown accelerations. Note that the same schemes can be reformulated expressing velocity and acceleration fields in function of the unknown displacements if* $\beta \neq 0$ *or expressing displacement and acceleration fields in function of the unknown velocities if* $\gamma \neq 0$.

So, integrating System (1.4) in time and using the approximations in Equations (1.10) and (1.11), one obtains N systems of algebraic equations, so that $\forall n \in \{0, \dots, N-1\}$:

$$\begin{array}{ll} \mathbb{M}\, {}^{n+1}\mathbf{A} + \mathbf{F}^{\text{int}}({}^{n+1}\mathbf{U}) = {}^{n+1}\mathbf{F}^{\text{ext}} & \text{in } \Omega^h|_{t^{n+1}} \\ {}^{n+1}\mathbf{U} = {}^{n}\mathbf{U} + {}^{n}\mathbf{V}\Delta t + {}^{n}\mathbf{A}\left(\frac{1}{2} - \beta\right)\Delta t^2 + {}^{n+1}\mathbf{A}\beta\Delta t^2 & \text{in } \Omega^h|_{t^{n+1}} \\ {}^{n+1}\mathbf{V} = {}^{n}\mathbf{V} + {}^{n}\mathbf{A}(1-\gamma)\Delta t + {}^{n+1}\mathbf{A}\gamma\Delta t & \text{in } \Omega^h|_{t^{n+1}} \\ {}^{n+1}\mathbf{U} = \overline{\mathbf{U}} & \text{over } \partial\Omega_u|_{t^{n+1}} \\ \{{}^{0}\mathbf{U}, {}^{0}\mathbf{V}\} = \{\mathbf{U}_0, \mathbf{V}_0\} & \text{in } \Omega^h|_{t_{\text{in}}} \end{array} \tag{1.12}$$

Varying the two parameters γ and β, System (1.12) requires different resolution strategies and denotes different properties. If the parameter $\beta = 0$, internal forces at the current time t^{n+1} are function of quantities computed in the previous time t^n only and the scheme is called *explicit*. On the contrary, if $\beta > 0$, internal forces at the current time t^{n+1} are function of the unknown current accelerations ${}^{n+1}\mathbf{A}$ and the scheme is called *implicit*. The main resolution schemes are reported in Table 1.1.

scheme	*type*	γ	β	*accuracy order*
Purely Explicit	explicit	0	0	1
Central Difference	explicit	1/2	0	2
Fox-Goodwin	implicit	1/2	1/12	2
Linear Acceleration	implicit	1/2	1/6	2
Average Acceleration	implicit	1/2	1/4	2

Table 1.1: Newmark's time integration schemes.

The parameter $\gamma = 1/2$ guarantees a second-order accuracy. In the undamped case, the stability is unconditioned if $2\beta \geq \gamma \geq 1/2$ or it is conditioned to:

$$\omega^h \Delta t \leq \frac{1}{\sqrt{\gamma/2 - \beta}} \tag{1.13}$$

for all the eigenfrequencies ω^h in the system, if $\gamma \geq 1/2$ and $\beta < \gamma/2$.

Remark 2 *For efficiency reasons, two types of scheme are mainly used: the explicit schemes that are conditionally stable, as the Central Difference scheme in Table 1.1, and the implicit schemes that are unconditionally stable, allowing the use of large time steps, like the Average Acceleration scheme in Table 1.1. Because of this, the Fox-Goodwin and Linear Acceleration schemes in Table 1.1, which are both implicit and conditionally stable, are not usually employed.*

Remark 3 *The Central Difference scheme is also equivalent to the Verlet Velocity algorithm [Swope et al., 1982], a derivation of the Verlet integration scheme [Verlet, 1967] developed for molecular dynamics.*

Remark 4 *When only the low mode response is of interest, as stated in [Hilber et al., 1977], it is advantageous for an algorithm to possess numerical dissipation, in order to damp out the spurious participation of the higher modes. The Newmark family allows to control the amount of dissipation fixing* $\beta = \left(\gamma + \frac{1}{2}\right)^2/4$ *and increasing* $\gamma > 1/2$. *In this way, the critical time step is increased but the second-order accuracy is lost. Other families of time integration methods, such as Houbolt's method [Houbolt, 1950] or Wilson* θ*-method [Bathe and Wilson, 1973], possess superior dissipative properties, with the downside of affecting the accuracy of the solution. The HHT algorithm is so introduced as an alternative in [Hilber et al., 1977] providing the Newmark's formulation with a parameter* α *exploited to calibrate the numerical dissipation:*

$$\mathbb{M}\,{}^{n+1}\mathbf{A} + \mathbf{F}^{int}({}^{n+\alpha}\mathbf{U}) = {}^{n+1}\mathbf{F}^{ext} \tag{1.14}$$

with ${}^{n+\alpha}\mathbf{U} = (1+\alpha)\,{}^{n+1}\mathbf{U} - \alpha\,{}^{n}\mathbf{U}$ *and* $\alpha \in [-1/3, 0]$.

3.1 Central Difference scheme

Following the Newmark's notation of the Central Difference scheme with $\gamma = 1/2$ and $\beta = 0$, System (1.12) reads $\forall n \in \{0, \dots, N-1\}$:

$$\begin{array}{ll}
\mathbf{M}\,{}^{n+1}\mathbf{A} = {}^{n+1}\mathbf{F}^{\text{ext}} - \mathbf{F}^{\text{int}}({}^{n+1}\mathbf{U}) & \text{in } \Omega^h|_{t^{n+1}} \\
{}^{n+1}\mathbf{U} = {}^{n}\mathbf{U} + {}^{n}\mathbf{V}\Delta t + {}^{n}\mathbf{A}\frac{\Delta t^2}{2} & \text{in } \Omega^h|_{t^{n+1}} \\
{}^{n+1}\mathbf{V} = {}^{n}\mathbf{V} + {}^{n}\mathbf{A}\frac{\Delta t}{2} + {}^{n+1}\mathbf{A}\frac{\Delta t}{2} & \text{in } \Omega^h|_{t^{n+1}} \\
{}^{n+1}\mathbf{U} = \overline{\mathbf{U}} & \text{over } \partial\Omega_u|_{t^{n+1}} \\
\{{}^{0}\mathbf{U}, {}^{0}\mathbf{V}\} = \{\mathbf{U}_0, \mathbf{V}_0\} & \text{in } \Omega^h|_{t_{\text{in}}}
\end{array} \tag{1.15}$$

where $\mathbf{M}$ is the lumped mass matrix, that is preferable to the consistent one $\mathbb{M}$ because it increases the stable time step and because it leads to an efficient resolution of System (1.15), in which the single degrees of freedom are computed solving single independent equations. Such system of independent equations can be then solved with high performance architectures and shared-memory multi-processing loops. As listed in Section 7.3.2 of [Hughes, 1987], the lumped mass matrix $\mathbf{M}$ can be obtained using:

- the nodal quadrature rules;
- the row-sum technique;
- the *Hinton-Rock-Zienkiewicz* (HRZ) technique: only the diagonal terms of the consistent mass matrix $\mathbb{M}$ are computed and then scaled so as to preserve the total element mass.

Each one of the mentioned technique presents advantages and disadvantages, as summarized in Table 1.2. Nodal quadrature rules and the row-sum technique applied to high-order elements determine negative or zero values at certain nodes. On the other hand,

lumping technique	*disadvantages*	*examples*
nodal quadrature	negative or zero values	axis nodes of the axisymmetric elements
		corner nodes of the Serendipity 8-node quadrilateral element
row-sum	negative or zero values	
HRZ	weak mathematical support	/

Table 1.2: Mass lumping techniques specified in [Hughes, 1987].

the HRZ technique avoids such values but has not been so far supported by mathematical proofs.

Nevertheless, the applications to simple elements like linear triangles or bi-linear quadrilaterals produce similar (if not identical) lumped mass matrices $\mathbf{M}$, without negative or zero values. The Abaqus/Explicit element library, for instance, offers only linear shape functions with reduced integration. In the Matlab prototypes of this work, the row-sum technique is adopted, as well as the choice to use bi-linear quadrilateral elements.

Furthermore, in case of contact and wave-propagation problems, e.g. impact problems, the Central Difference scheme is generally the preferable direct integration procedure, also because the current nodal displacements ${}^{n+1}\mathbf{U}$ depend on nodal quantities computed in the previous time step at t^n, the current configuration does not change with the algebraic computations of the unknown accelerations ${}^{n+1}\mathbf{A}$ and the nonlinearities, as in contact problems, do not require iterative computations of the solution.

On the other hand, the explicit time integration schemes are conditionally stable. As stated in Equation (1.13), the time step choice is governed by the eigenfrequencies of the problem. So, a stable time step Δt_{stable} for an undamped problem is estimated as:

$$\Delta t_{\text{stable}} \leq \frac{2}{\omega_{\text{max}}} \tag{1.16}$$

where ω_{max} is the maximum eigenfrequency of the spatial discretization Ω^h. It can be demonstrated that:

$$\omega_{\text{max}} \leq \max_{\text{el}}\{\omega_{\text{el}}\} \tag{1.17}$$

and ω_{el} are the eigenfrequencies of the individual elements “el” of the mesh Ω^h.

Only undamped systems are considered in this work. Even if common sense would suggest one to think differently, viscous damping reduces the stable time step, so that:

$$\Delta t_{\text{stable}} \leq \frac{2}{\omega^{\text{max}}}\left(\sqrt{1+\zeta^2}-\zeta\right) \tag{1.18}$$

where ω_{max} is again the maximum eigenfrequency of the undamped system and ζ is the fraction of critical damping in the highest mode.

Analytical expressions for bounding the maximum eigenfrequency $\omega_{\max}$ were given in [Flanagan and Belytschko, 1985] for the case of reduced-integration quadrilateral and hexahedral elements with undistorted or distorted shapes.

Furthermore, a physical interpretation of the stability limit was introduced as *Courant-Friedrichs-Lewy* (CFL) condition [Courant et al., 1967], which prescribes that the stable time step Δt_{stable} must be smaller than the time required by a dilatational stress wave (that is always bigger than an eventual shear wave) to traverse the smallest element of the given mesh Ω^h:

$$\Delta t_{\text{stable}} \leq \frac{L_{\text{el}}}{c} \tag{1.19}$$

where L_{el} is the characteristic length of the element and c is the wave propagation speed.

The characteristic length is differently estimated depending on the element technology. Some examples are summarized in table 1.3 from the *LS-Dyna* theoretical manual [Hallquist, 1998], where the estimations are defined considering the efficiency of their calculation.

element technology	*characterist length* L_{el}
2-node truss and beam	length
3-node triangle and 4-node tetrahedron	minimum altitude
4-node quadrilateral and 8-node hexaedron	$V_{\text{el}}/A_{\text{el,max}}$ [1]

Table 1.3: Characteristic length estimations for the main element technologies.

For elastic materials with constant bulk modulus, Young's modulus E, Poisson's coefficient ν and density ρ, the dilatational wave propagation speeds are summarized in table 1.4 in relation to the main model assumptions.

model assumption	*dilatational wave speed c*
truss and beam	$\sqrt{E/\rho}$
plane stress and shell	$\sqrt{E/(\rho(1-\nu)^2)}$
solid and plane strain	$\sqrt{(E(1-\nu))/(\rho(1+\nu)(1-2\nu))}$

Table 1.4: Wave speeds with the main model assumptions.

For the sake of simplicity, in the majority of the analyses of this work, a constant time step is chosen *a priori*, verifying at the beginning of the analysis:

$$\Delta t \leq 0.8 \frac{V_{\text{el}}}{A_{\text{el,max}} c} \tag{1.20}$$

because of the choice of quadrilateral bi-linear elements.

---

[1] V_{el} is the volume of the element "el", $A_{\text{el,max}} = \max_i\{A_{i,\text{facet}}\}$ is the biggest facet area of the element "el".

So, considering a constant time step Δt, the solution at each time t^{n+1} is computed in four sequential steps:

$$\begin{array}{ll} \text{1. update of displacements:} & {}^{n+1}\mathbf{U} = {}^{n}\mathbf{U} + {}^{n}\mathbf{V}\Delta t + {}^{n}\mathbf{A}\frac{\Delta t^2}{2} \\ \text{2. update of mid-step velocities:} & {}^{n+\frac{1}{2}}\mathbf{V} = {}^{n}\mathbf{V} + {}^{n}\mathbf{A}\frac{\Delta t}{2} \\ \text{3. accelerations resolution:} & \mathbf{M}\,{}^{n+1}\mathbf{A} = {}^{n+1}\mathbf{F}^{\text{ext}} - {}^{n+1}\mathbf{F}^{\text{int}} \\ \text{4. update of velocities:} & {}^{n+1}\mathbf{V} = {}^{n+\frac{1}{2}}\mathbf{V} + {}^{n+1}\mathbf{A}\frac{\Delta t}{2} \end{array} \tag{1.21}$$

where the internal forces ${}^{n+1}\mathbf{F}^{\text{int}}$ are computed in function of displacements ${}^{n+1}\mathbf{U}$ and the eventual viscous forces could be computed in function of mid-step velocities ${}^{n+\frac{1}{2}}\mathbf{V}$.

Remark 5 *Note that the extension of the 4-step solution sequence* (1.21) *to an adaptive time step scheme as in [Hibbitt and Karlsson, 1979] is straightforward: the time step should be chosen before the step 1 and* Δt^n *remains fixed in the sequence. Nonetheless, the velocities* ${}^{n+1}\mathbf{V}$ *in step 4 are computed only for output purposes and the explicit commercial codes are used to reduce the 4-step solution sequence* (1.21) *to a 3-step solution sequence, replacing the step 4 velocity definition in the first 2 steps of the following sequence. In this case, the use of an adaptive time step scheme involves two time step quantities* Δt^{ave} *and* Δt^n *as follows:*

$$\begin{array}{ll} \textit{1. update of displacements:} & {}^{n+1}\mathbf{U} = {}^{n}\mathbf{U} + {}^{n-\frac{1}{2}}\mathbf{V}\Delta t^n + {}^{n}\mathbf{A}\Delta t^n \Delta t^{ave} \\ \textit{2. update of mid-step velocities:} & {}^{n+\frac{1}{2}}\mathbf{V} = {}^{n-\frac{1}{2}}\mathbf{V} + {}^{n}\mathbf{A}\Delta t^{ave} \\ \textit{3. accelerations resolution:} & \mathbf{M}\,{}^{n+1}\mathbf{A} = {}^{n+1}\mathbf{F}^{ext} - {}^{n+1}\mathbf{F}^{int} \end{array} \tag{1.22}$$

where $\Delta t^{ave} = (\Delta t^{n-1} + \Delta t^n)/2$ *as defined in the time line in Figure 1.1.*

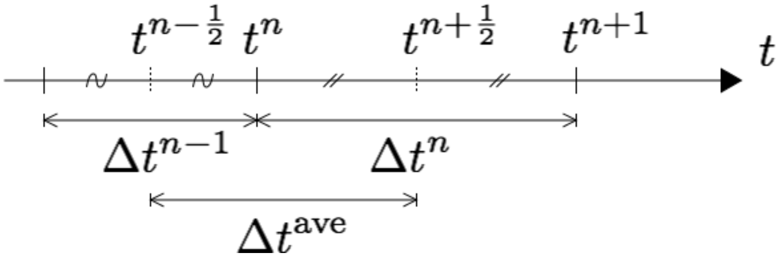


Figure 1.1: Definition of times $t^{n-\frac{1}{2}}$, t^n, $t^{n+\frac{1}{2}}$, t^{n+1} and time steps Δt^n, Δt^{n+1}, Δt^{ave}.

3.2 Alternative schemes

The parameters choice of the Central Difference scheme does not introduce any numerical dissipation. Spurious oscillations can so be detected for highly transient dynamic cases, as wave propagation problems. The introduction of a non-diagonal damping matrix should be then necessary to attenuate or avoid such spurious oscillations, requiring a matrix factorization to solve the system. Other explicit time integration schemes have been formulated for introducing numerical dissipation. For instance, the extension of the same *HHT* algorithm (briefly described in Remark 4) has been formulated and studied in [Chung and Lee, 1994] and [Hulbert and Chung, 1996]. The so-called Tchamwa-Wielgosz scheme represents another interesting approach, introduced in [Tchamwa et al., 1999] as purely explicit scheme (with $\gamma = 0$) and so only first-order accurate. Comparative studies in [Rio et al., 2005] have then shown that remaining dissipative explicit schemes are second-order accurate but often provide less accurate solutions than the Tchamwa-Wielgosz scheme. So, the research of accurate time integration schemes for wave propagation analysis is still ongoing. In what follows, two potential strategies are presented.

3.2.1 Two-steps explicit time integration

A two-steps explicit time integration scheme is formulated and shown to be stable and accurate in [Noh and Bathe, 2013]. Such integration scheme is based on the splitting of the time step in two sub-steps in order to keep a non-diagonal damping matrix always on the right-hand side of the solving systems. Five parameters are then introduced to optimize the stability and accuracy of calculation.

Introducing the non-diagonal viscous damping matrix $\mathbb{C}$ and the first parameter $p \in (0,1)$ as marker of the position within the time step, the solution in terms of accelerations, velocities and displacements in the first sub-step reads:

$$\begin{array}{ll}
\text{1. update of mid-step displacements:} & {}^{n+p}\mathbf{U} = {}^{n}\mathbf{U} + {}^{n}\mathbf{V} p\Delta t + {}^{n}\mathbf{A} p^2 \frac{\Delta t^2}{2} \\
\text{2. update of mid-step velocities:} & {}^{n+p}\tilde{\mathbf{V}} = \left({}^{n}\mathbf{V} + {}^{n}\mathbf{A} p\frac{\Delta t}{2}\right)(1-s) + {}^{n}\mathbf{V} s \\
\text{3. accelerations resolution:} & \mathbf{M}\,{}^{n+p}\mathbf{A} = {}^{n+p}\mathbf{F}^{\text{ext}} - \mathbf{F}^{\text{int}}\left({}^{n+p}\mathbf{U}\right) - \mathbb{C}\,{}^{n+p}\tilde{\mathbf{V}} \\
\text{4. update of mid-step velocities:} & {}^{n+p}\mathbf{V} = {}^{n}\mathbf{V} + \left({}^{n}\mathbf{A} + {}^{n+p}\mathbf{A}\right) p\frac{\Delta t}{2}
\end{array} \tag{1.23}$$

where s is a second parameter, necessary only in the damped case.

The second sub-step of the scheme then reads:

$$\begin{array}{ll}
\text{1. update of displacements:} & {}^{n+1}\mathbf{U} = {}^{n+p}\mathbf{U} + {}^{n+p}\mathbf{V}(1-p)\Delta t + {}^{n+p}\mathbf{A}(1-p)^2 \frac{\Delta t^2}{2} \\
\text{2. update of velocities:} & {}^{n+1}\tilde{\mathbf{V}} = \left({}^{n+p}\mathbf{V} + {}^{n+p}\mathbf{A}(1-p)\frac{\Delta t}{2}\right)(1-s) + {}^{n+p}\mathbf{V} s \\
\text{3. accelerations resolution:} & \mathbf{M}\,{}^{n+1}\mathbf{A} = {}^{n+1}\mathbf{F}^{\text{ext}} - \mathbf{F}^{\text{int}}\left({}^{n+1}\mathbf{U}\right) - \mathbb{C}\,{}^{n+1}\tilde{\mathbf{V}} \\
\text{4. update of velocities:} & {}^{n+1}\mathbf{V} = {}^{n+p}\mathbf{V} + {}^{n+p}\mathbf{A}(1-p)\frac{\Delta t}{2} + \\
 & \quad + \left({}^{n}\mathbf{A} q_0 + {}^{n+p}\mathbf{A} q_1 + {}^{n+1}\mathbf{A} q_2\right)(1-p)\Delta t
\end{array} \tag{1.24}$$

where q_0, q_1 and q_2 are additional parameters.

Note that the damping matrix $\mathbb{C}$ is always in the right-hand side of the accelerations resolution.

The five parameters are calibrated to obtain a second-order accuracy, so that:

$$q_0 + q_1 + q_2 = \frac{1}{2}, \qquad q_2 = \frac{1}{2} - pq_1, \qquad s = -1 \tag{1.25}$$

and to obtain stability, so that:

$$q_1 = \frac{1-2p}{2p(1-p)}, \qquad 0.5 \leq p \leq 2 - \sqrt{2}. \tag{1.26}$$

In conclusion, the parameter p is recommended in [Noh and Bathe, 2013] to a value of 0.54 with further accuracy studies. Indeed, with such parameter value, in the low frequency response, the period elongation is about 1% and the amplitude decay is about 2.5% when the time step is 20% of the natural period or smaller. Furthermore, for a 2-dimensional wave propagation problem, the dispersion error is shown to be less than 2% and the higher frequencies are cut out of the solution.

So, such integration time scheme allows to use a viscous damping matrix $\mathbb{C}$ without affecting the accuracy and the resolution strategies of explicit time schemes via the inversion of a lumped mass matrix $\mathbf{M}$. On the other hand, 8 operation steps (reducible to 6) are required for the resolution of a single time step, against the 3 operation steps in Equation (1.22).

3.2.2 Multi-field space-time integration

In contrast to explicit time integration schemes that aim at minimizing the computational effort for the systems resolution, a multi-fields space-time monolithic discretization was proposed in [Hulbert and Hughes, 1990], exploiting a Time-Discontinuous Galerkin method (TDG) in order to consider jumps in the velocity field, as in wave-propagation problems.

The formulation is based on the integration of the weak formulation of the initial-boundary value problem in System (1.1) both in space and time as:

$$\begin{aligned} \int_{t_{\text{in}}}^{t_{\text{end}}} \int_{\Omega} (\rho \ddot{\mathbf{u}} \cdot \delta\dot{\mathbf{u}} + \boldsymbol{\sigma}(\mathbf{u}) : \boldsymbol{\varepsilon}(\delta\dot{\mathbf{u}}))\ d\Omega dt = \int_{t_{\text{in}}}^{t_{\text{end}}} \int_{\Omega} \mathbf{f}_{\text{body}} \cdot \delta\dot{\mathbf{u}}\ d\Omega dt + \\ + \int_{t_{\text{in}}}^{t_{\text{end}}} \int_{\partial\Omega_f} \mathbf{f}_{\text{surf}} \cdot \delta\dot{\mathbf{u}}\ d\Gamma dt \qquad \forall \delta\mathbf{u} = \mathbf{0} \text{ over } \partial\Omega_u \times [t_{\text{in}}, t_{\text{end}}] \end{aligned} \tag{1.27}$$

and the displacement solution is monolithically discretized in space and time, so that:

$$\mathbf{u}(\mathbf{x}, t) = (\boldsymbol{\phi}(\mathbf{x}) \times \boldsymbol{\psi}(t)) : \mathbb{U} \tag{1.28}$$

where $\boldsymbol{\psi}(t)$ denotes a piecewise-continuous time discretization and $\mathbb{U}$ contains the whole solution in terms of displacements in the overall space-time domain $\Omega \times [t_{\text{in}}, t_{\text{end}}]$, such that $\mathbb{U} = [{}^{0}\mathbf{U}, \ldots, {}^{N}\mathbf{U}]$, in which N is the total number of time steps according to the chosen time discretization. In this way, considering homogeneous boundary conditions and null initial values, the weak formulation in Equation (1.27) directly returns:

$$\left(\mathbb{K} \times \left(\mathbb{Q}^{10} + \mathbb{B}^{00}\right) + \mathbb{M} \times \left(\mathbb{Q}^{12} + \mathbb{B}^{11}\right)\right) : \mathbb{U} = \left(\mathbf{I} \times \mathbb{Q}^{10}\right) : \mathbb{F} \tag{1.29}$$

where $\mathbb{K}$ and $\mathbb{M}$ are the stiffness and consistent mass matrices, respectively, $\mathbf{I}$ is the identity matrix, $\mathbb{F}$ denotes the external forces history, $\mathbb{Q}^{ij}$ denotes the integration of the time model over the continuous part and $\mathbb{B}^{ij}$ denotes the discontinuous time model, i and j being in both cases the time derivative orders, such that:

$$
\begin{aligned}
\mathbb{Q}^{10} &= \int_{t_{\text{in}}}^{t_{\text{end}}} \dot{\boldsymbol{\psi}} \times \boldsymbol{\psi}\, dt, \qquad \mathbb{Q}^{12} = \int_{t_{\text{in}}}^{t_{\text{end}}} \dot{\boldsymbol{\psi}} \times \ddot{\boldsymbol{\psi}}\, dt, \\
\mathbb{B}^{00} &= \boldsymbol{\psi}(0) \times \boldsymbol{\psi}(0) + \sum_{n=1}^{N} \left(\boldsymbol{\psi}(t_+^n) \times \boldsymbol{\psi}(t_+^n) - \boldsymbol{\psi}(t_-^{n+1}) \times \boldsymbol{\psi}(t_-^{n+1}) \right), \\
\mathbb{B}^{11} &= \dot{\boldsymbol{\psi}}(0) \times \dot{\boldsymbol{\psi}}(0) + \sum_{n=1}^{N} \left(\dot{\boldsymbol{\psi}}(t_+^n) \times \dot{\boldsymbol{\psi}}(t_+^n) - \dot{\boldsymbol{\psi}}(t_-^{n+1}) \times \dot{\boldsymbol{\psi}}(t_-^{n+1}) \right).
\end{aligned}
\tag{1.30}
$$

The extension of such displacement-based formulation in Equation (1.28) to a two-fields formulation can be advantageous: both displacements and velocities become unknowns and different time discretizations $\boldsymbol{\psi}_u(t)$ and $\boldsymbol{\psi}_v(t)$ can be considered (e.g. continuous for displacements and piecewise-continuous for velocities). The continuity between velocities and displacement first derivative is then weakly imposed. So, discretizing with:

$$
\begin{aligned}
\mathbf{u}(\mathbf{x},t) &= (\boldsymbol{\phi}(\mathbf{x}) \times \boldsymbol{\psi}_u(t)) : \mathbb{U} \\
\mathbf{v}(\mathbf{x},t) &= (\boldsymbol{\phi}(\mathbf{x}) \times \boldsymbol{\psi}_v(t)) : \mathbb{V}
\end{aligned}
\tag{1.31}
$$

the final system becomes:

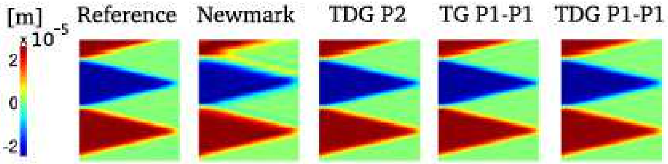
$$
\begin{bmatrix} \mathbb{K} \times \left(\mathbb{Q}_{uu}^{01} + \mathbb{B}_{uu}^{00} \right) & -\mathbb{K} \times \mathbb{Q}_{uv}^{00} \\ \mathbb{K} \times \mathbb{Q}_{vu}^{00} & \mathbb{M} \times \left(\mathbb{Q}_{vv}^{01} + \mathbb{B}_{vv}^{00} \right) \end{bmatrix} : \begin{bmatrix} \mathbb{U} \\ \mathbb{V} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \left(\mathbf{I} \times \mathbb{Q}_{vv}^{00} \right) : \mathbb{F} \end{bmatrix}. \tag{1.32}
$$

The size of the coefficient matrix in Equation (1.32) is significant, multiplying the number of displacement and velocity degrees of freedom by the number of time steps in the time discretization. So, advanced reduced order modeling technologies have been employed in [Boucinha et al., 2013] and [Boucinha et al., 2014] in order to resolve the expensive monolithic systems with acceptable computational costs. Accuracy gains are evidenced in both the papers. Figures 1.2, taken from [Boucinha et al., 2013], show the results of a 2-dimensional example comparing the exact reference solution with:

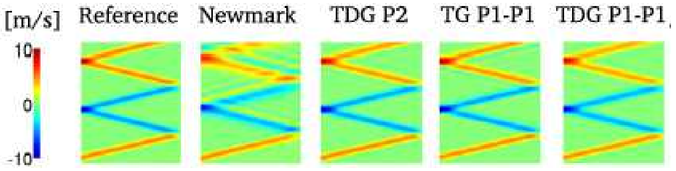
- displacement-based elements and average accelerations Newmark' scheme;
- displacement-based elements and quadratic TDG;
- multi-field-based elements and linear time shape functions (TG);
- multi-field-based elements and linear TDG.

Figures 1.2 so highlight a better accuracy of the multi-field TDG approach, where most of the spurious oscillations are removed.

The extension to nonlinear problems remains however an important research topic for the robust utilization in industrial cases.



(a) displacements



(b) velocities

Figure 1.2: Results comparison for a 2-dimensional wave propagation problem taken from [Boucinha et al., 2013].

3.3 Energy-Momentum integration scheme for large rotations

The initial-boundary value problem has been so far considered assuming the linear elastic formulation presented in Equations (1.1) involving only translational displacements $\mathbf{u}(\mathbf{x},t)$. Out of a more accurate formulation, as stated in [Simo and Tarnow, 1992], temporal and spatial Finite Element discretizations of the continuum dynamics need not, and in general will not, inherit the conservation of momentum properties and the a-priori estimate. For instance, the conservation form of the mid-point rule is an exact momentum conserving algorithm which does not conserve energy for autonomous Hamiltonian systems, except for the linear regime.

In [Simo and Tarnow, 1992], a so-called Energy-Momentum (EM) integration scheme was introduced for geometrical non-linearities of quadratic nature, aiming at preserving specific features of the continuous system such as conservation of momentum, angular momentum and energy when the system and the applied forces allow to. The extensions to nonlinear dynamics of shells and rods were later proposed in [Simo and Tarnow, 1994] and [Simo et al., 1995], where displacement-based discretizations are used, and in [Sansour et al., 2002], where a multi-field discretization is treated involving enhanced-strain Finite Elements (see the theoretical basis of the Enhanced Assumed Strain method for linear elastic problems in Appendix B).

Defining the kinetic energy $\mathcal{T}_{\text{shell}}$ of the shell space as in [Sansour et al., 2002], one gets:

$$\mathcal{T}_{\text{shell}} = \int_{\Gamma_{\text{shell}}} \left(\rho h \dot{\mathbf{u}} \cdot \dot{\mathbf{u}} + \frac{1}{12} \rho h^3 \dot{\mathbf{w}} \cdot \dot{\mathbf{w}} \right) d\Gamma \qquad (1.33)$$

where Γ_{shell} is the shell surface, h is the shell thickness, $\mathbf{u}$ denotes the displacement field, computed as difference between current and reference middle surface positions, and $\mathbf{w}$ indicates the difference vector, defined as the difference between current and reference forward-rotated directors.

The formulation of the internal energy, considering an elastic material behavior, can be shortly written as:

$$\mathcal{V}_{\text{shell}} = \int_{\Gamma_{\text{shell}}} \left(V^1(\mathbf{E}^0(\mathbf{u},\mathbf{w})) + V^2(\mathbf{K}(\mathbf{u},\mathbf{w})) \right) \, d\Gamma \tag{1.34}$$

where $\mathbf{E}^0$ and $\mathbf{K}$ are the main strain measures of the shell defined in [Sansour et al., 2002].

As shown in [Simo and Tarnow, 1992], defining the kinematical fields at the n^{th} time step as:

$$^{n+\xi}\mathbf{u} = {^n}\mathbf{u}\,(1-\xi) + {^{n+1}}\mathbf{u}\,\xi, \quad {^{n+\xi}}\mathbf{w} = {^n}\mathbf{w}\,(1-\xi) + {^{n+1}}\mathbf{w}\,\xi, \tag{1.35}$$

the associated time derivatives as:

$$\begin{aligned} {^{n+\xi}}\dot{\mathbf{u}} &= \frac{^{n+1}\mathbf{u} - {^n}\mathbf{u}}{\Delta t}, \quad & {^{n+\xi}}\dot{\mathbf{w}} &= \frac{^{n+1}\mathbf{w} - {^n}\mathbf{w}}{\Delta t}, \\ {^{n+\xi}}\ddot{\mathbf{u}} &= \frac{^{n+1}\dot{\mathbf{u}} - {^n}\dot{\mathbf{u}}}{\Delta t}, \quad & {^{n+\xi}}\ddot{\mathbf{w}} &= \frac{^{n+1}\dot{\mathbf{w}} - {^n}\dot{\mathbf{w}}}{\Delta t} \end{aligned} \tag{1.36}$$

and the strain measures as:

$$^{n+\xi}\mathbf{E}^0 = {^n}\mathbf{E}^0\,(1-\xi) + {^{n+1}}\mathbf{E}^0\,\xi, \quad {^{n+\xi}}\mathbf{K} = {^n}\mathbf{K}\,(1-\xi) + {^{n+1}}\mathbf{K}\,\xi, \tag{1.37}$$

taking $\xi = 1/2$, the following equation holds:

$$^{n}\mathcal{T}_{\text{shell}} + {^n}\mathcal{V}_{\text{shell}} + {^n}\mathcal{F}^{\text{ext}} = {^{n+1}}\mathcal{T}_{\text{shell}} + {^{n+1}}\mathcal{V}_{\text{shell}} + {^{n+1}}\mathcal{F}^{\text{ext}} \tag{1.38}$$

which means that the conservation of energy is assured and where $\mathcal{F}^{\text{ext}}$ indicates the energy associated to a conservative loading.

Further developments presented in [Sansour et al., 2002], show that adding the incompatible enhanced strains $\tilde{\mathbf{E}}$ to the strain measure $\mathbf{E}^0$ so that:

$$\mathcal{V}_{\text{shell}} = \int_{\Gamma_{\text{shell}}} \left(V^1(\mathbf{E}^0(\mathbf{u},\mathbf{w}) + \tilde{\mathbf{E}}) + V^2(\mathbf{K}(\mathbf{u},\mathbf{w})) \right) \, d\Gamma \tag{1.39}$$

and discretizing:

$$^{n+\xi}\tilde{\mathbf{E}} = {^n}\tilde{\mathbf{E}}\,(1-\xi) + {^{n+1}}\tilde{\mathbf{E}}\,\xi \tag{1.40}$$

keeping $\xi = 1/2$, the energy is still conserved.

The extension to general constitutive models was proposed in [Laursen and Meng, 2001], whereas rotational formulations for hyper-elastic models were proposed for rods and shells in [Pimenta et al., 2008] and [Campello et al., 2011], respectively.

4 Multi-scale coupling strategies in space and time

The need to couple different models and discretizations arises when treating large structures with localized phenomena, e.g. small details, micro-cracks, defects, highly localized loading conditions, structured material and so on. In this case, a refined and specialized model should be focused on a restricted zone of the overall domain. In the rest of the domain, not involved in particular phenomena, such as in the case of regular geometry, smoothly distributed loading and homogeneous material, the model should be as coarse as possible, in order to save computing time.

Hereafter, the problem in Figure 1.3 will be referred to as the *reference problem*. It consists of a sample structure in a domain Ω composed of two parts. The part on the left, characterized by fine features, requires a fine resolution both in space and time. The part on the right requires only a coarse resolution both in space and time.

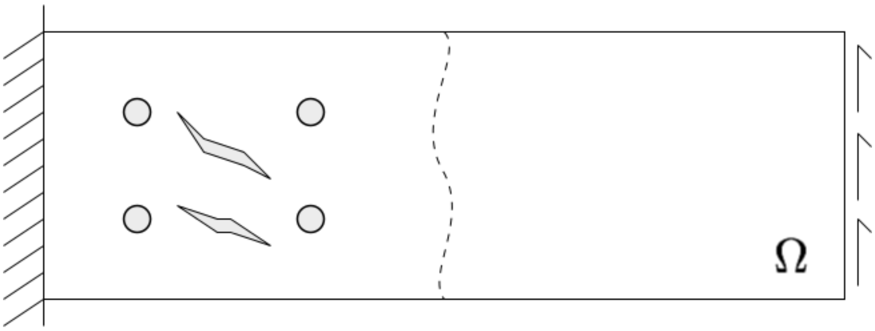


Figure 1.3: Localized cracks and holes in a sample structure.

4.1 Partitioning in time for a given non-uniform mesh

A potential methodology to solve the problem in Figure 1.3 consists in discretizing the domain with a non-uniform mesh Ω^h, for instance pulling together two different uniform meshes Ω_1^h and Ω_2^h, so that $\Omega_1^h \cup \Omega_2^h \equiv \Omega^h$ and treating the embedded meshes over the interface $\Gamma^h \equiv \Omega_1^h \cap \Omega_2^h$ using a tie constraint. Two different time integration schemes or time steps are then applied in the two partitions Ω_1^h and Ω_2^h according to the two different mesh sizes.

For instance, Figure 1.4 shows the partitioning of the overall domain in a fine mesh Ω_1^h composed of 20 quadrilateral elements and a coarse mesh Ω_2^h composed of just 1 quadrilateral element. The common nodes between the two regions constitute the interface Γ^h.

In literature, several works have been devoted to algorithms for coupling implicit and explicit time integration schemes using the same time scale everywhere. These approaches are usually referred to as Mixed-Methods in time. On the other hand, several other works have been exploring the possibility of associating different time scales, developing a multi-scale approach in time, usually called Subcycling technique.

4.1.1 Mixed-Methods (implicit-explicit couplings)

The possibility to couple explicit and implicit Newmark's integration schemes with same time step was proposed in [Hughes and Liu, 1978a] based on an element partition of the

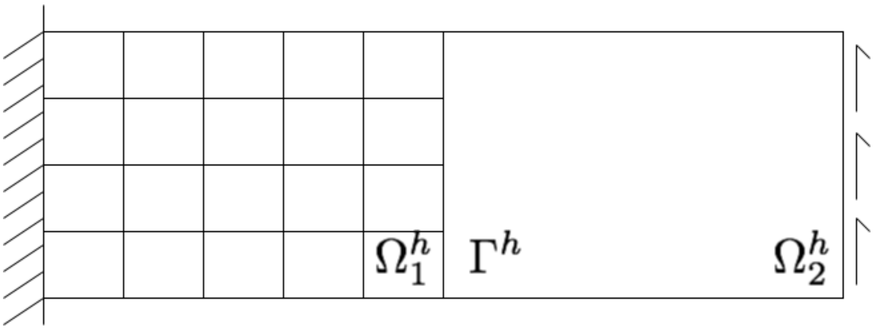


Figure 1.4: Non-uniform mesh as combination of two uniform meshes.

domain and using the same time step for all the parts, so that a linear elastic problem could be solved by:

$$\begin{aligned}&\left(\mathbf{M}+\mathbb{K}_I\beta_I\Delta t^2\right){}^{n+1}\mathbf{A} = {}^{n+1}\mathbf{F}^{\text{ext}}+\\ &-\mathbb{K}_I\left({}^{n}\mathbf{U}+{}^{n}\mathbf{V}\Delta t+{}^{n}\mathbf{A}\left(\frac{1}{2}-\beta_I\right)\Delta t^2\right)-\mathbb{K}_E\left({}^{n}\mathbf{U}+{}^{n}\mathbf{V}\Delta t+{}^{n}\mathbf{A}\frac{\Delta t^2}{2}\right)\end{aligned} \tag{1.41}$$

where the subscripts I and E refer to the implicit and explicit partitions, respectively.

A nodal partitioning of the mesh was then proposed in [Belytschko and Mullen, 1978], having the system of degrees of freedoms subdivided into explicit and implicit blocks, with:

$$\mathbf{M}=\begin{bmatrix}\mathbf{M}_E & \mathbf{0}\\ \mathbf{0} & \mathbf{M}_I\end{bmatrix},\quad \mathbb{K}=\begin{bmatrix}\mathbb{K}_{E|E} & \mathbb{K}_{E|I}\\ \mathbb{K}_{I|E} & \mathbb{K}_{I|I}\end{bmatrix}, \tag{1.42}$$

solving at first the explicit partition:

$${}^{n+1}\mathbf{U}_E = 2\,{}^{n}\mathbf{U}_E - {}^{n-1}\mathbf{U}_E + \mathbf{M}_E^{-1}\left({}^{n+1}\mathbf{F}_E^{\text{ext}} - \mathbb{K}_{E|E}{}^{n}\mathbf{U}_E - \mathbb{K}_{E|I}{}^{n}\mathbf{U}_I\right)\Delta t^2 \tag{1.43}$$

and then the implicit one:

$$\begin{aligned}&\left(\mathbf{M}_I+\mathbb{K}_{I|I}\beta_I\Delta t^2\right){}^{n+1}\mathbf{A}_I = {}^{n+1}\mathbf{F}_I^{\text{ext}} - \mathbb{K}_{I|E}{}^{n+1}\mathbf{U}_E+\\ &-\mathbb{K}_{I|I}\left({}^{n}\mathbf{U}_I+{}^{n}\mathbf{V}_I\Delta t+{}^{n}\mathbf{A}_I\left(1/2-\beta_I\right)\Delta t^2\right).\end{aligned} \tag{1.44}$$

4.1.2 Subcycling technique

In the Subcycling technique formulated in [Belytschko et al., 1979] for different time steps, also different coupling schemes were proposed: implicit-implicit, implicit-explicit and, finally, explicit-explicit couplings. The crucial issue was the time interpolation over the interface Γ^h between the partitions. The initial methods used a linear interpolation. These were shown to be stable for first order systems in [Belytschko et al., 1985]. Further studies were devoted to nodal partitioning stability in [Smolinski et al., 1988] or to error estimation in [Zienkiewicz and Xie, 1991].

The extension to second order systems was then proposed in [Daniel, 1997], with an explicit Subcycling algorithm considering an integer ratio between the time steps and proving that the linear interpolation leads to narrow bands of instability. These can be eliminated either by the addition of artificial viscosity, or by other Subcycling methods, which are more complex. Stable Subcycling methods for explicit-explicit couplings in second order systems were also introduced in [Smolinski et al., 1996] and [Daniel, 2003].

In particular, the algorithm in [Daniel, 2003] was based on a so-called *partial velocity algorithm*. Considering the domain partition in Figure 1.4 and two different pre-fixed time steps Δt_1 and Δt_2 in Ω_1^h and Ω_2^h, respectively, so that $\Delta t_2 = M\Delta t_1$, such algorithm computed at first the solution at the n^{th} coarse time step with:

$$\begin{aligned} &\mathbf{M}\,{}^{n}\mathbf{A}_{2,\text{in}} = {}^{n}\mathbf{F}^{\text{ext}}_{2,\text{in}} - \mathbb{K}_{2,\text{in}|2}{}^{n}\mathbf{U}_2 && \text{in } \Omega^h_{2,\text{in}} \\ &\mathbf{M}\,{}^{n}\mathbf{A}_{2,\Gamma} = {}^{n}\mathbf{F}^{\text{ext}}_{2,\Gamma} - \mathbb{K}_{2,\Gamma|2}{}^{n}\mathbf{U}_2 + {}^{n}\mathbf{F}_{\Gamma} && \text{over } \Gamma^h \\ &{}^{n+\frac{1}{2}}\mathbf{V}_2 = {}^{n-\frac{1}{2}}\mathbf{V}_2 + {}^{n+1}\mathbf{A}_2\Delta t_2 && \text{in } \Omega^h_2 \end{aligned} \tag{1.45}$$

where the subscript "in" denotes the nodes in the partition $\Omega^h_{2,\text{in}} \equiv \Omega^h_2 \setminus \Gamma^h$, the subscript "$\Gamma$" indicates the nodes over the interface Γ^h, ${}^{n}\mathbf{F}_{\Gamma}$ denotes the forces acting across the interface Γ^h and contains zero values at the the nodes in the rest of the domain $\Omega^h_{2,\text{in}} \equiv \Omega^h_2 \setminus \Gamma^h$, the lumped matrix $\mathbf{M}$ is assembled considering the overall domain $\Omega^h \equiv \Omega^h_1 \cup \Omega^h_2$, the stiffness matrix $\mathbb{K}$ is assembled considering the two partitions as completely separated and velocities are updated as in the 3-step algorithm of Equation (1.22). The solution in the finer partition Ω^h_1 is then computed in the finer time stepping, so that $\forall m \in \{1, \dots, M\}$:

$$\begin{aligned} &\mathbf{M}\,{}^{m}\mathbf{A}_{1,\text{in}} = {}^{m}\mathbf{F}^{\text{ext}}_{1,\text{in}} - \mathbb{K}_{1,\text{in}|1}{}^{m}\mathbf{U}_1 && \text{in } \Omega^h_{1,\text{in}} \\ &\mathbf{M}\,{}^{m}\mathbf{A}_{1,\Gamma} = {}^{m}\mathbf{F}^{\text{ext}}_{1,\Gamma} - \mathbb{K}_{1,\Gamma|1}{}^{m}\mathbf{U}_1 - {}^{n}\mathbf{F}_{\Gamma} && \text{over } \Gamma^h \\ &{}^{m+\frac{1}{2}}\mathbf{V}_1 = {}^{m-\frac{1}{2}}\mathbf{V}_1 + {}^{m}\mathbf{A}_1\Delta t_1 && \text{in } \Omega^h_1 \end{aligned} \tag{1.46}$$

assuming that the forces across the interface Γ^h are constant in the finer time stepping. The displacements at nodes not belonging to the interface Γ^h are then updated with the Central Difference scheme in the the normal manner, whereas over the interface Γ^h:

$${}^{m+1}\mathbf{U}_{\Gamma} = {}^{m+1}\mathbf{U}_{\Gamma} + \left({}^{m+\frac{1}{2}}\mathbf{V}_{1,\Gamma} + {}^{n+\frac{1}{2}}\mathbf{V}_{2,\Gamma}\right)\Delta t_1 \tag{1.47}$$

where the partial velocities are computed with partial accelerations ignoring the interface forces ${}^{n}\mathbf{F}_{\Gamma}$.

Despite the easy extension to adaptive time steps and rotational degrees of freedom and advantages for a wide range of problems, the Subcycling technique reveals issues of robustness in contact problems and of efficiency with constraints as ties or fasteners.

4.2 Domain Decomposition method

The Domain Decomposition method provides a way for partitioning a given domain $\Omega \subset \mathbb{R}^{\mathcal{N}}$ in many subdomains $\Omega_i \subset \mathbb{R}^{\mathcal{N}}$ solved independently and connected together via interfacial conditions. In this description, the given domain in Figure 1.3 is divided into two subdomains Ω_1 and Ω_2, such that $\Omega_1 \cup \Omega_2 \equiv \Omega$.

The Domain Decomposition methods are divided in two classes:

- the overlapping Domain Decomposition methods, in which the subdomains are partially or completely overlapped so that $\Omega_1 \cap \Omega_2 \equiv \Omega_{12} \subset \mathbb{R}^{\mathcal{N}}$;
- the non-overlapping Domain Decomposition methods, in which the subdomains are overlapped only across an interface Γ of a lower dimension with respect to the subdomains, so that $\Omega_1 \cap \Omega_2 = \Gamma \subset \mathbb{R}^{\mathcal{N}-1}$.

The non-overlapping Domain Decomposition methods (see [Gosselet and Rey, 2006] for an extended overview) are suitable for mechanical interpretations. Basically, the notion of interface Γ can be linked to a set of conservation principles and phenomenological laws: e.g. the conservation of fluxes imposes the mechanical equilibrium over an interface, or again the contact laws enable the disjunction of subdomains but prohibit interpenetration between them, whereas a perfect interface supposes that the displacement field (or pressure or temperature) is continuous.

In this context, two main methods arose in parallel: the Balancing Domain Decomposition (BDD) [LeTallec et al., 1991] and [Mandel, 1993] and the Finite Element Tearing and Interconnecting (FETI) [Farhat and Roux, 1991]. The BDD is usually referred to as *primal* approach and consists in choosing the interface displacement field as main unknown and solving iteratively equilibrium and force conditions. On the contrary, the FETI is a *dual* approach and consists in privileging the interface force conditions and satisfying the equilibrium and the displacement interface continuity with an iterative process.

Primal and *dual* approaches were efficiently extended to highly heterogeneous systems of equations via cheap substructure-based preconditioners in [Rixen and Farhat, 1999] and [Rixen, 2002] and applying more accurate initial estimations of interface forces in [Gosselet et al., 2003]. Multi-scale non-overlapping Domain Decomposition methods were proposed in [Zohdi and Wriggers, 1999] and [Zohdi et al., 2001] for treating arbitrary geometries, loadings and heterogeneous micro-structures, as a way of material volume-based sub-structuring.

A linear combination of the interface displacement and force fields, so-called *mixed* approach, is sometimes preferred to both the methods BDD and FETI. An artificial stiffness parameter is introduced over the interface, so that a null stiffness is equivalent to a *dual* approach and an infinite stiffness produces a *primal* approach. A *mono*-scale Domain Decomposition method was proposed in [Ladevžee, 1985], in which all unknowns over the interfaces are introduced. The multi-scale version was then presented for static problems in [Dureisseix and Ladevèze, 1998] and [Ladevèze et al., 2001] and extended to multiple scales in time in [Ladevèze and Nouy, 2003]. Further recent applications are proposed in [Cresta et al., 2007] and [Hinojosa et al., 2014] for buckling and post-buckling analyses, respectively, in [Guidault et al., 2007] for fracture problems, in [Kerfriden et al., 2009] and [Saavedra et al., 2012] for debonding and delamination in laminates, respectively. *Mixed* approaches also enable to provide a more natural framework to handle complex interfaces, such as contact and friction, than classical approaches, as proposed in [Champaney et al., 1997]. An extension to contact dynamics has been more recently presented in [Odièvre et al., 2010].

4.2.1 FETI in explicit dynamics

In the FETI approach, extended to transient dynamics in [Farhat et al., 1994], the continuity and the equilibrium between subdomains is enforced through the use of Lagrange multipliers $\boldsymbol{\lambda}$.

The weak formulation of Problem (1.1), splitting the domain Ω into two subdomains Ω_1 and Ω_2, reads in the subdomain Ω_1:

$$\begin{aligned}&\int_{\Omega_1} (\rho\ddot{\mathbf{u}}_1 \cdot \delta\mathbf{u}_1 + \boldsymbol{\sigma}(\mathbf{u}_1) : \boldsymbol{\varepsilon}(\delta\mathbf{u}_1))\ d\Omega + \int_\Gamma \boldsymbol{\lambda} \cdot \delta(\mathbf{w}_1 - \mathbf{w}_2)\ d\Gamma = \\ &= \int_{\Omega_1} \mathbf{f}_{\text{body}} \cdot \delta\mathbf{u}_1\ d\Omega + \int_{\partial\Omega_f} \mathbf{f}_{\text{surf}} \cdot \delta\mathbf{u}_1\ d\Gamma \qquad \forall \delta\mathbf{u}_1 = \mathbf{0} \text{ over } \partial\Omega_u \times [t_{\text{in}}, t_{\text{end}}]\end{aligned} \tag{1.48}$$

in the subdomain Ω_2:

$$\begin{aligned}&\int_{\Omega_2} (\rho\ddot{\mathbf{u}}_2 \cdot \delta\mathbf{u}_2 + \boldsymbol{\sigma}(\mathbf{u}_2) : \boldsymbol{\varepsilon}(\delta\mathbf{u}_2))\ d\Omega + \int_\Gamma \boldsymbol{\lambda} \cdot \delta(\mathbf{w}_1 - \mathbf{w}_2)\ d\Gamma = \\ &= \int_{\Omega_2} \mathbf{f}_{\text{body}} \cdot \delta\mathbf{u}_2\ d\Omega + \int_{\partial\Omega_f} \mathbf{f}_{\text{surf}} \cdot \delta\mathbf{u}_2\ d\Gamma \qquad \forall \delta\mathbf{u}_2 = \mathbf{0} \text{ over } \partial\Omega_u \times [t_{\text{in}}, t_{\text{end}}]\end{aligned} \tag{1.49}$$

and over the interface Γ:

$$\int_\Gamma \delta\boldsymbol{\lambda} \cdot (\mathbf{w}_1 - \mathbf{w}_2)\ d\Gamma = 0 \qquad \forall \delta\boldsymbol{\lambda} \tag{1.50}$$

where $\mathbf{w}$ is the kinematic quantity (displacement, velocity or acceleration) to be constrained over the interface Γ.

Applying the displacement-based Finite Element spatial discretization with conforming meshes over the interface, so that $\Gamma^h \equiv \Gamma_1^h \equiv \Gamma_2^h$, and applying the Central Difference scheme with same time steps for both the subdomains, Equations (1.48), (1.49) and (1.50) become $\forall n \in \{0, \ldots, N-1\}$:

$$\begin{aligned}&\mathbf{M}_1{}^{n+1}\mathbf{A}_1 + \mathbf{C}_1^T\ {}^{n+1}\boldsymbol{\Lambda} = {}^{n+1}\mathbf{F}_1^{\text{ext}} - {}^{n+1}\mathbf{F}_1^{\text{int}} && \text{in } \Omega_1^h|_{t^{n+1}} \\ &\mathbf{M}_2{}^{n+1}\mathbf{A}_2 - \mathbf{C}_2^T\ {}^{n+1}\boldsymbol{\Lambda} = {}^{n+1}\mathbf{F}_2^{\text{ext}} - {}^{n+1}\mathbf{F}_2^{\text{int}} && \text{in } \Omega_2^h|_{t^{n+1}} \\ &\mathbf{C}_1{}^{n+1}\mathbf{W}_1 = \mathbf{C}_2{}^{n+1}\mathbf{W}_2 && \text{over } \Gamma^h|_{t^{n+1}}\end{aligned} \tag{1.51}$$

where matrices $\mathbf{C}_i$ are Boolean unitary matrices that extract nodal values from the subdomains to the interfaces:

$$\mathbf{C}_{i,(\dim\Gamma_i^h \times \dim\Omega_i^h)} : \ \Omega_i^h \mapsto \Gamma_i^h \quad i = 1,2 \tag{1.52}$$

and vectors $\mathbf{W}_i$ denote the kinematic quantities to be constrained over the interface Γ^h.

The notation:

$${}^{n+1}\mathbf{W}_{i,\Gamma} \triangleq \mathbf{C}_i{}^{n+1}\mathbf{W}_i \quad i = 1,2 \tag{1.53}$$

will be employed in what follows.

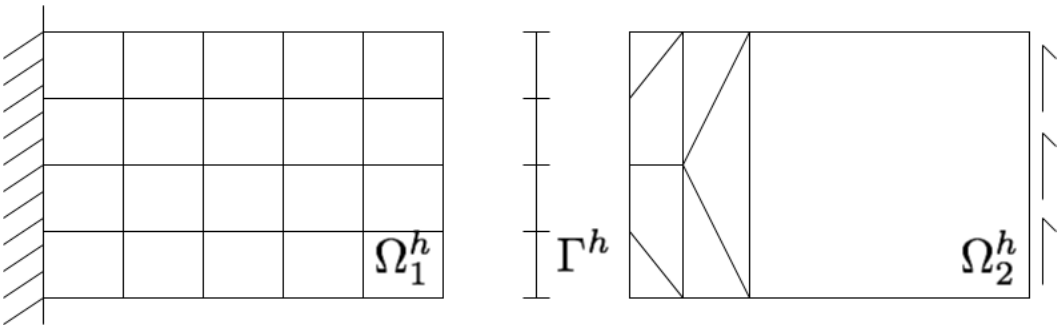


Figure 1.5: Domain Decomposition into two subdomains with conforming meshes.

As stated in [Combescure and Gravouil, 2001] and [Combescure and Gravouil, 2002], these kinematic quantities are restricted to accelerations $\mathbf{A}_i$ or velocities $\mathbf{V}_i$ for using explicit time integration schemes in the couplings.

Figure 1.5 shows an example of spatial discretization with conforming meshes, with the employment of additional triangle elements to divide the coarse subdomain Ω_2^h.

Nevertheless, constraining the subdomains meshes to be conforming across the interface Γ^h is not desirable. It is possible to adapt the method to be able to apply arbitrary meshes in the subdomains as in in Figure 1.6 via the spatial interpolation of the Lagrange multipliers over the discretized interfaces. So, System (1.54) becomes $\forall n \in \{0,\ldots,N-1\}$:

$$\begin{array}{ll} \mathbf{M}_1{}^{n+1}\mathbf{A}_1 + \mathbf{C}_1^T\,{}^{n+1}\mathbf{\Lambda} = {}^{n+1}\mathbf{F}_1^{\text{ext}} - {}^{n+1}\mathbf{F}_1^{\text{int}} & \text{in } \Omega_1^h|_{t^{n+1}} \\ \mathbf{M}_2{}^{n+1}\mathbf{A}_2 - \mathbf{C}_2^T\Pi_h^{1T}\,{}^{n+1}\mathbf{\Lambda} = {}^{n+1}\mathbf{F}_2^{\text{ext}} - {}^{n+1}\mathbf{F}_2^{\text{int}} & \text{in } \Omega_2^h|_{t^{n+1}} \\ {}^{n+1}\mathbf{W}_{1,\Gamma} = \Pi_h^1\,{}^{n+1}\mathbf{W}_{2,\Gamma} & \text{over } \Gamma_1^h|_{t^{n+1}} \end{array} \tag{1.54}$$

where Π_h^1 and Π_h^{1T} are the space down-scaling and up-scaling operators, respectively:

$$\begin{array}{l} \Pi^1_{h,(\dim\Gamma_1^h \times \dim\Gamma_2^h)} : \Gamma_2^h \mapsto \Gamma_1^h, \\ \Pi^{1T}_{h,(\dim\Gamma_2^h \times \dim\Gamma_1^h)} : \Gamma_1^h \mapsto \Gamma_2^h. \end{array} \tag{1.55}$$

4.2.2 FETI and Subcycling coupling

Coupling embedded interface meshes as in Figure 1.6 with explicit time integration schemes naturally leads to coupling multiple time steps. Two different time discretizations impose the problem to choose which quantities should be interpolated and in which way.

Here, the first subdomain Ω_1^h has a fine uniform mesh and a fine time step $\Delta t_1 = t^{m+1} - t^m$, whereas the second one Ω_2^h has a coarse uniform mesh and a coarse time step $\Delta t_2 = t^{n+1} - t^n$. The ratio between the two time steps is equivalent to the ratio between the two mesh sizes, so that $\Delta t_2 = M\Delta t_1$ and $t_{\text{end}} - t_{\text{in}} = N\Delta t_2$. Figure 1.7 shows the time discretization applied to the two subdomains. The difference between the time steps is

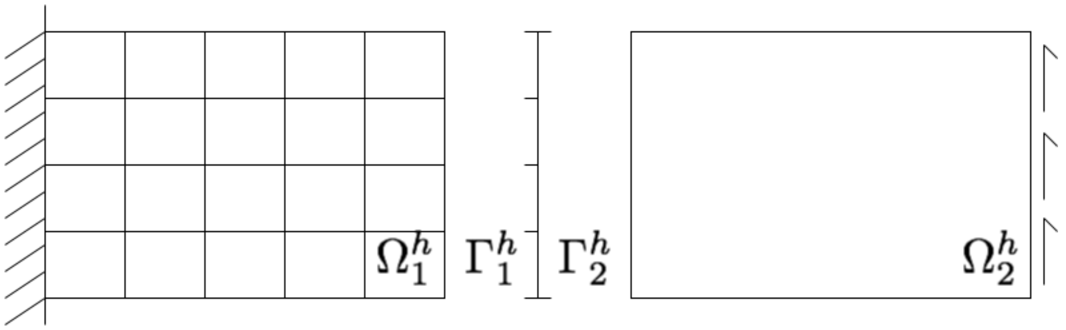


Figure 1.6: Domain Decomposition into two subdomains with embedded interface meshes.

usually treated with a linear temporal interpolation operator $\Pi_t^1\left({}^{n}\square, {}^{n+1}\square\right)$ of quantity $\square$, so that:

$$ {}^{m}\square = {}^{n}\square\left(1-\frac{m}{M}\right) + {}^{n+1}\square\left(\frac{m}{M}\right) \quad \forall m \in \{0,\ldots,M\}. \tag{1.56}$$

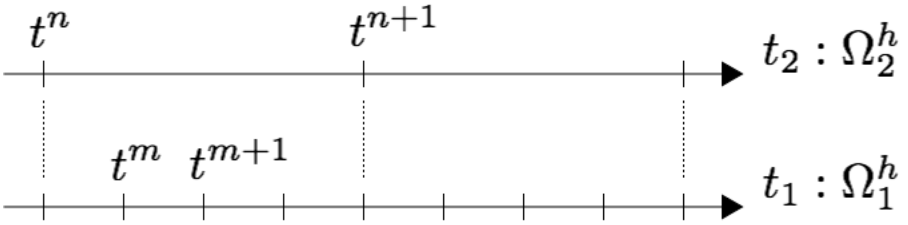


Figure 1.7: Time discretization for the two subdomains with different mesh sizes.

4.2.3 Multi-time-step FETI approach

An algorithm for the resolution of System (1.54) in the case of multiple time steps was introduced in [Gravouil and Combescure, 2001]. Here, the velocities $\mathbf{V}_i$ are selected as kinematic quantities $\mathbf{W}_i$ to be constrained over the interface Γ_1^h, in order to obtain the stability properties stated in [Gravouil and Combescure, 2001]. System (1.54) so becomes $\forall n \in \{0,\ldots,N-1\}$:

$$\begin{array}{ll} \mathbf{M}_1{}^{m+1}\mathbf{A}_1 + \mathbf{C}_1^T\,{}^{m+1}\boldsymbol{\Lambda} = {}^{m+1}\mathbf{F}_1^{\text{ext}} - {}^{m+1}\mathbf{F}_1^{\text{int}} & \text{in } \Omega_1^h|_{t^{m+1}}\ \forall m \in \{0,\ldots,M-1\} \\ \mathbf{M}_2{}^{n+1}\mathbf{A}_2 - \mathbf{C}_2^T\Pi_h^{1T}\,{}^{n+1}\boldsymbol{\Lambda} = {}^{n+1}\mathbf{F}_2^{\text{ext}} - {}^{n+1}\mathbf{F}_2^{\text{int}} & \text{in } \Omega_2^h|_{t^{n+1}} \\ {}^{m+1}\mathbf{V}_{1,\Gamma} = \Pi_h^1\Pi_t^1\left({}^{n}\mathbf{V}_{2,\Gamma}, {}^{n+1}\mathbf{V}_{2,\Gamma}\right) & \text{over } \Gamma_1^h|_{t^{m+1}}\ \forall m \in \{0,\ldots,M-1\}. \end{array} \tag{1.57}$$

The basic idea of the algorithm is to split the kinematical solution of each subdomain into two terms, respectively denoted as *free* (or *unconstrained*) and *link* (or *constrained*) problems. The first one refers to the free motion of each subdomain under the effect of the external loads and boundary conditions. The second one evaluates the correction to the first solution, computing the interface forces to be applied for "re-gluing" the subdomains.

Note that for explicit-explicit couplings, because of the mass lumping, the second solution concerns only the degrees of freedom of the interface nodes over the interface Γ^h.

The *free* problem reads $\forall n \in \{0,\dots,N-1\}$:

$$\begin{aligned} \mathbf{M}_1{}^{m+1}\mathbf{A}_1^{\text{free}} &= {}^{m+1}\mathbf{F}_1^{\text{ext}} - {}^{m+1}\mathbf{F}_1^{\text{int}} \quad \text{in } \Omega_1^h|_{t^{m+1}}\ \forall m \in \{0,\dots,M-1\} \\ \mathbf{M}_2{}^{n+1}\mathbf{A}_2^{\text{free}} &= {}^{n+1}\mathbf{F}_2^{\text{ext}} - {}^{n+1}\mathbf{F}_2^{\text{int}} \quad \text{in } \Omega_2^h|_{t^{n+1}} \end{aligned} \tag{1.58}$$

while the *link* one $\forall n \in \{0,\dots,N-1\}$:

$$\begin{aligned} \mathbf{M}_{1,\Gamma}{}^{m+1}\mathbf{A}_{1,\Gamma}^{\text{link}} &= -{}^{m+1}\mathbf{\Lambda} \quad \text{over } \Gamma_1^h|_{t^{m+1}}\ \forall m \in \{0,\dots,M-1\} \\ \mathbf{M}_{2,\Gamma}{}^{n+1}\mathbf{A}_{2,\Gamma}^{\text{link}} &= \Pi_h^{1T}\ {}^{n+1}\mathbf{\Lambda} \quad \text{over } \Gamma_2^h|_{t^{n+1}} \end{aligned} \tag{1.59}$$

and the interface constraint is developed splitting the velocities in *free* and *link* terms. The third equation of System (1.57) becomes $\forall n \in \{0,\dots,(N-1)\}$:

$$\begin{aligned} {}^{m+1}\mathbf{V}_{1,\Gamma}^{\text{free}} + {}^{m+1}\mathbf{V}_{1,\Gamma}^{\text{link}} &= \Pi_h^1\ {}^{n}\mathbf{V}_{2,\Gamma}^{\text{free}}\left(1-\frac{m}{M}\right) + \Pi_h^1\ {}^{n+1}\mathbf{V}_{2,\Gamma}^{\text{free}}\left(\frac{m}{M}\right) + \\ &+ \Pi_h^1\ {}^{n}\mathbf{V}_{2,\Gamma}^{\text{link}}\left(1-\frac{m}{M}\right) + \Pi_h^1\ {}^{n+1}\mathbf{V}_{2,\Gamma}^{\text{link}}\left(\frac{m}{M}\right) \\ &\text{over } \Gamma_1^h|_{t^{m+1}} \forall m \in \{0,\dots,M-1\} \end{aligned} \tag{1.60}$$

assuming that:

$$\begin{aligned} {}^{m}\mathbf{V}_2^{\text{free}} &= {}^{n}\mathbf{V}_2^{\text{free}}\left(1-\frac{m}{M}\right) + {}^{n+1}\mathbf{V}_2^{\text{free}}\left(\frac{m}{M}\right) \\ {}^{m}\mathbf{V}_{2,\Gamma}^{\text{link}} &= {}^{n}\mathbf{V}_{2,\Gamma}^{\text{link}}\left(1-\frac{m}{M}\right) + {}^{n+1}\mathbf{V}_{2,\Gamma}^{\text{link}}\left(\frac{m}{M}\right) \end{aligned} \tag{1.61}$$

and:

$$\begin{aligned} {}^{m+1}\mathbf{V}_i^{\text{free}} &= {}^{m}\mathbf{V}_i + {}^{m}\mathbf{A}_i\frac{\Delta t_i}{2} + {}^{m+1}\mathbf{A}_i^{\text{free}}\frac{\Delta t_i}{2} \quad i = 1,2, \\ {}^{m+1}\mathbf{V}_{i,\Gamma}^{\text{link}} &= {}^{m+1}\mathbf{A}_{i,\Gamma}^{\text{link}}\frac{\Delta t_i}{2} \quad i = 1,2. \end{aligned} \tag{1.62}$$

In this way, five sets of unknowns must be computed for each coarse time step, as summarized in Algorithm 1.

```
while $n \le N$ do
    1. free coarse solution: second of Equations (1.58);
    while $m \le M$ do
        2. free fine solution: first of Equations (1.58);
        3. interface constraint: Equation (1.60);
        4. link fine solution: first of Equations (1.59);
        update $m \leftarrow m + 1$;
    end
    5. link coarse solution: second of Equations (1.59);
    update $n \leftarrow n + 1$;
end
```

Algorithm 1: GC algorithm scheme.

The acceleration solution is then obtained as sum between *free* and *link* solutions:

$$ {}^{m+1}\mathbf{A}_i = {}^{m+1}\mathbf{A}_i^{\text{free}} + {}^{m+1}\mathbf{A}_i^{\text{link}} \quad i = 1,2. \tag{1.63}$$

More details about the *GC* algorithm will be given in Chapter 2.

4.2.4 Application of the energy method to the multi-time-step coupling

As pointed out in [Gravouil and Combescure, 2001] and [Mahjoubi et al., 2009], the *GC* algorithm imposing the continuity of velocities over the interfaces is a stable algorithm, because it leads to energy unbalances always dissipative. An algebraic study is conducted in [Gravouil and Combescure, 2001] with the use of the energy method introduced in [Hughes and Liu, 1978b] for explicit-implicit couplings.

Considering external forces constant in time and defining:

- the variation operators $[\Box] = {}^{n+1}\Box - {}^{n}\Box$ and $\llbracket\Box\rrbracket = \sum_{m=0}^{M-1}\left({}^{m+1}\Box - {}^{m}\Box\right)$ for the coarse and the fine time stepping, respectively,
- the term related to the kinetic energy as ${}^nT = \frac{1}{2}{}^n\mathbf{A}^T\left(\mathbf{M} - \mathbb{K}\Delta t^2/4\right){}^n\mathbf{A}$,
- the term related to the internal energy as ${}^nV = \frac{1}{2}{}^n\mathbf{V}^T\mathbb{K}\,{}^n\mathbf{V}$,
- the term related to the interface energy as ${}^nE_\Gamma = \frac{1}{\Delta t}\left[{}^n\mathbf{V}_\Gamma\right]^T\left[{}^n\mathbf{\Lambda}\right]$

the energy method states that a given coupling method is stable if:

$$\llbracket {}^nT_1 \rrbracket + [{}^nT_2] + \llbracket {}^nV_1 \rrbracket + [{}^nV_2] \leq 0 \quad \forall n \in \{1,\ldots,N\} \tag{1.64}$$

with:

$$\llbracket {}^nT_1 \rrbracket + [{}^nT_2] + \llbracket {}^nV_1 \rrbracket + [{}^nV_2] = {}^nE_\Gamma = {}^nE_{\Gamma_1} + {}^nE_{\Gamma_2}. \tag{1.65}$$

Developing the definition of ${}^nE_\Gamma$, one can write:

$${}^nE_\Gamma = \frac{1}{\Delta t_1}\llbracket {}^n\mathbf{V}_{1,\Gamma} \rrbracket^T \llbracket {}^n\mathbf{\Lambda} \rrbracket + \frac{1}{\Delta t_2}[{}^n\mathbf{V}_{2,\Gamma}]^T[{}^n\mathbf{\Lambda}]. \tag{1.66}$$

It is demonstrated then in Section 2 of [Gravouil and Combescure, 2001] that the inequality in Equation (1.64) is always verified with the *GC* algorithm and that, in presence of different time scales in the subdomains, there is always numerical dissipation, due to the linear interpolation of the interface velocities in time. The calculation of such dissipation in function of the subcycling parameters (e.g. the ratio between the different time steps M) was given in [Prakash and Hjelmstad, 2004] and [Mahjoubi et al., 2009].

Based upon the energy method application, further works that aimed at developing energy-preserving couplings, as the *PH* algorithm introduced in [Prakash and Hjelmstad, 2004] with the generic Newmark's formulation or the *MGC* algorithm described in [Mahjoubi et al., 2011] based on an incremental formulation but extended to other time integration schemes have been proposed.

4.2.5 Energy-preserving multi-time-step FETI approach

As asserted in [Prakash and Hjelmstad, 2004], a change in the interfacial constraint is necessary to avoid numerical dissipation in presence of different time scales. The idea is to impose the velocity continuity once per coarse time step, defining Lagrange multipliers in order to verify the condition ${}^{n}E_{\Gamma} = 0 \ \forall n \in \{1,\dots,N\}$.

The equilibrium at each fine time step is verified through the definition of the *free unbalanced* reaction force ${}^{m}\mathbf{S}^{\text{free}}$ and the *link unbalanced* reaction force ${}^{m}\mathbf{S}^{\text{link}}$ in function of the coarse *free* and *link* quantities, respectively, linearly interpolated in time.

In contrast to Equation (1.61) in the *GC* algorithm, the linear interpolation in time is different for *free* and *link* problems:

$$\begin{aligned} {}^{m}\mathbf{A}_2^{\text{free}} &= {}^{n}\mathbf{A}_2\left(1-\frac{m}{M}\right)+{}^{n+1}\mathbf{A}_2^{\text{free}}\left(\frac{m}{M}\right)\\ {}^{m}\mathbf{A}_2^{\text{link}} &= {}^{n+1}\mathbf{A}_2^{\text{link}}\left(\frac{m}{M}\right). \end{aligned} \tag{1.67}$$

The *free unbalanced* reaction force ${}^{m}\mathbf{S}^{\text{free}}$ and the *link unbalanced* reaction force ${}^{m}\mathbf{S}^{\text{link}}$ read, respectively:

$$\begin{aligned} {}^{m}\mathbf{S}^{\text{free}} = \Pi_h^1 &\left(\left({}^{n}\mathbf{F}^{\text{ext}}_{2,\Gamma} - {}^{n}\mathbf{F}^{\text{int}}_{2,\Gamma}\right)\left(1-\frac{m}{M}\right) + \left({}^{n+1}\mathbf{F}^{\text{ext}}_{2,\Gamma} - {}^{n+1}\mathbf{F}^{\text{int}}_{2,\Gamma}\right)\left(\frac{m}{M}\right)\right) + \\ &-\Pi_h^1\mathbf{M}_{2,\Gamma}\left({}^{n}\mathbf{A}_{2,\Gamma}\left(1-\frac{m}{M}\right)+{}^{n+1}\mathbf{A}^{\text{free}}_{2,\Gamma}\left(\frac{m}{M}\right)\right) \end{aligned} \tag{1.68}$$

and:

$${}^{m}\mathbf{S}^{\text{link}} = -\Pi_h^1\mathbf{M}_{2,\Gamma}{}^{n+1}\mathbf{A}^{\text{link}}_{2,\Gamma}\left(\frac{m}{M}\right) - {}^{m}\mathbf{\Lambda}. \tag{1.69}$$

The latter, using the Lagrange multipliers definition in Equation (1.59), becomes:

$${}^{m}\mathbf{S}^{\text{link}} = {}^{n+1}\mathbf{\Lambda}\left(\frac{m}{M}\right) - {}^{m}\mathbf{\Lambda}. \tag{1.70}$$

The definitions in Equations (1.68) and (1.69) lead to the following properties:

$${}^{m}\mathbf{S}^{\text{free}} + {}^{m}\mathbf{S}^{\text{link}} = \mathbf{0} \quad \forall m \in \{1,\dots,M\},\ {}^{n}\mathbf{S}^{\text{free}} = {}^{n}\mathbf{S}^{\text{link}} = \mathbf{0} \quad \forall n \in \{1,\dots,N\}. \tag{1.71}$$

Combining Equation (1.70) with the first of Equations (1.71), the Lagrange multipliers to be applied to the fine subdomain Ω_1^h are:

$${}^{m}\mathbf{\Lambda} = {}^{m}\mathbf{S}^{\text{free}} + {}^{n+1}\mathbf{\Lambda}\left(\frac{m}{M}\right). \tag{1.72}$$

In this way, the *free* problem reads $\forall n \in \{0,\dots,N-1\}$:

$$\begin{aligned} &\mathbf{M}_1{}^{m+1}\mathbf{A}_1^{\text{free}} = {}^{m+1}\mathbf{F}_1^{\text{ext}} - {}^{m+1}\mathbf{F}_1^{\text{int,free}} - \mathbf{C}_1^{T}\ {}^{m+1}\mathbf{S}^{\text{free}} && \text{in } \Omega_1^h|_{t^{m+1}}\ \forall m \in \{0,\dots,M-1\}\\ &\mathbf{M}_2{}^{n+1}\mathbf{A}_2^{\text{free}} = {}^{n+1}\mathbf{F}_2^{\text{ext}} - {}^{n+1}\mathbf{F}_2^{\text{int}} && \text{in } \Omega_2^h|_{t^{n+1}} \end{aligned} \tag{1.73}$$

the *link* one $\forall n \in \{0,\ldots,N-1\}$:

$$\begin{array}{ll} \mathbf{M}_1{}^{m+1}\mathbf{A}_1^{\text{link}} = -\mathbf{C}_1^T\,{}^{n+1}\boldsymbol{\Lambda}\left(\frac{m}{M}\right) - {}^{m+1}\mathbf{F}_1^{\text{int,link}} & \text{in } \Omega_1^h|_{t^{m+1}}\ \forall m \in \{0,\ldots,M-1\} \\ \mathbf{M}_{2,\Gamma}{}^{n+1}\mathbf{A}_{2,\Gamma}^{\text{link}} = \Pi_h^{1T}\,{}^{n+1}\boldsymbol{\Lambda} & \text{over } \Gamma_2^h|_{t^{n+1}} \end{array} \tag{1.74}$$

and the interface constraint, in contrast to Equation (1.60) in the *GC* algorithm, is verified only at the coarse time step $\forall n \in \{0,\ldots,N-1\}$:

$$^{n+1}\mathbf{V}_{1,\Gamma}^{\text{free}} + {}^{n+1}\mathbf{V}_{1,\Gamma}^{\text{link}} = \Pi_h^1\,{}^{n+1}\mathbf{V}_{2,\Gamma}^{\text{free}} + \Pi_h^1\,{}^{n+1}\mathbf{V}_{2,\Gamma}^{\text{link}} \quad \text{over } \Gamma_1^h|_{t^{n+1}}. \tag{1.75}$$

Note that, differently from the *GC* algorithm, the *link* problem in Equation (1.74) here involves all the nodes of the finer subdomain.

It is proven in Section 5 of [Prakash and Hjelmstad, 2004] that such formulation leads to ${}^nE_\Gamma = 0\ \forall n \in \{1,\ldots,N\}$.

So, the *free* and *link* fine problems are solved in two sequential time loops before and after the interface constraint, respectively, as described in Algorithm 2.

```
while $n \leq N$ do
    1. free coarse solution: second of Equations (1.73);
    while $m \leq M$ do
        2. free fine solution: first of Equations (1.73);
        update $m \leftarrow m+1$;
    end
    3. interface constraint: Equation (1.75);
    while $m \leq M$ do
        4. link fine solution: first of Equations (1.74);
        update $m \leftarrow m+1$;
    end
    5. link coarse solution: second of Equations (1.74);
    update $n \leftarrow n+1$;
end
```

Algorithm 2: PH algorithm scheme.

The *GCbis* algorithm is equivalent to the *PH* algorithm, except the computation of the unbalanced reaction forces ${}^m\mathbf{S}^{\text{free}}$, that are substituted by the first part of the temporal linear interpolation of the Lagrange multipliers $\left(1-\frac{m}{M}\right)\boldsymbol{\Lambda}^n$, so that Equation (1.72) becomes:

$$^m\boldsymbol{\Lambda} = {}^n\boldsymbol{\Lambda}\left(1-\frac{m}{M}\right) + {}^{n+1}\boldsymbol{\Lambda}\left(\frac{m}{M}\right) \quad \forall m \in \{1,\ldots,M\} \tag{1.76}$$

that is a simple linear interpolation in time $\Pi_t^1\left({}^n\square, {}^{n+1}\square\right)$. This change is more practical and efficient and does not affect the energy-preserving property ${}^nE_\Gamma = 0\ \forall n \in \{1,\ldots,N\}$. Nonetheless, the extensions of the *PH* and *GCbis* algorithms to nonlinear problems and to adaptive time steps remains computationally expensive.

More details about the *GCbis* algorithm will be given in Chapter 2.

4.2.6 Non-overlapping Domain Decomposition methods for dynamic adaptivity

Non-overlapping Domain Decomposition methods constitute an advanced, robust and efficient technique for multi-scale couplings. Nevertheless, the industrial aim of this work is oriented towards an automatized evolution of the scales topology in order to follow evolutive mechanisms and optimize the use of a finer scale. Domain Decomposition would then require the support of a robust re-meshing strategy in order to change the position of the different subdomains during the analysis. Such implementation is considered to be intrusive in commercial software where the meshing strategy is included in a pre-processing package. However, the associated results have been taken as reference for the developments in this work.

4.3 Arlequin framework

The Arlequin method is an example of overlapping Domain Decomposition method, described for static problems in [BenDhia and Rateau, 2005] after previous works of the same authors. Two ways of partitioning the overall domain of the problem are possible:

- a completely overlapping method for coupling global and local domains in multi-model and multi-grid analyses,
- a partially overlapping method for coupling different subdomains positioned in different zones of the overall domain.

The method was extended to couplings between solids and molecular models in [Xiao and Belytschko, 2004] and [Xu and Belytschko, 2008] and to dynamics couplings in [Ghanem et al., 2013].

The coupling is based on the utilization of spatial weight functions η_i (for the kinetic energy term), α_i (for the internal energy term) and φ_i (for the external work term) in the classical Domain Decomposition weak formulation with two subdomains. In the subdomain Ω_1, one can write:

$$\begin{aligned}&\int_{\Omega_1} (\alpha_1\, \rho \ddot{\mathbf{u}}_1 \cdot \delta\mathbf{u}_1 + \eta_1\, \boldsymbol{\sigma}(\mathbf{u}_1) : \boldsymbol{\varepsilon}(\delta\mathbf{u}_1))\; d\Omega + C(\boldsymbol{\lambda}, \delta\mathbf{u}_1) = \\ &= \int_{\Omega_1} \varphi_1\, \mathbf{f}_{\text{body}} \cdot \delta\mathbf{u}_1\; d\Omega + \int_{\partial\Omega_f} \varphi_1\, \mathbf{f}_{\text{surf}} \cdot \delta\mathbf{u}_1\; d\Gamma \qquad \forall \delta\mathbf{u}_1 = \mathbf{0} \text{ over } \partial\Omega_u \times [t_{\text{in}}, t_{\text{end}}]\end{aligned} \tag{1.77}$$

whereas in the subdomain Ω_2:

$$\begin{aligned}&\int_{\Omega_2} (\alpha_2\, \rho \ddot{\mathbf{u}}_2 \cdot \delta\mathbf{u}_2 + \eta_2\, \boldsymbol{\sigma}(\mathbf{u}_2) : \boldsymbol{\varepsilon}(\delta\mathbf{u}_2))\; d\Omega - C(\boldsymbol{\lambda}, \delta\mathbf{u}_2) = \\ &= \int_{\Omega_2} \varphi_2\, \mathbf{f}_{\text{body}} \cdot \delta\mathbf{u}_2\; d\Omega + \int_{\partial\Omega_f} \varphi_2\, \mathbf{f}_{\text{surf}} \cdot \delta\mathbf{u}_2\; d\Gamma \qquad \forall \delta\mathbf{u}_2 = \mathbf{0} \text{ over } \partial\Omega_u \times [t_{\text{in}}, t_{\text{end}}]\end{aligned} \tag{1.78}$$

and in the overlapped zone Ω_{12}:

$$C(\delta\boldsymbol{\lambda}, \mathbf{w}_1 - \mathbf{w}_2) = 0 \qquad \forall \delta\boldsymbol{\lambda} \tag{1.79}$$

where $C(\square,\square)$ is the continuity operator and $\mathbf{w}$ is the constrained kinetic quantity. The weight functions are chosen so that $\sum_i \alpha_i = \sum_i \eta_i = \sum_i \varphi_i = 1$ is always verified in the overall domain Ω.

Figure 1.8 shows a potential application of the partially overlapping version of the Arlequin method and the associated shape of the weight functions varying only in the horizontal direction.

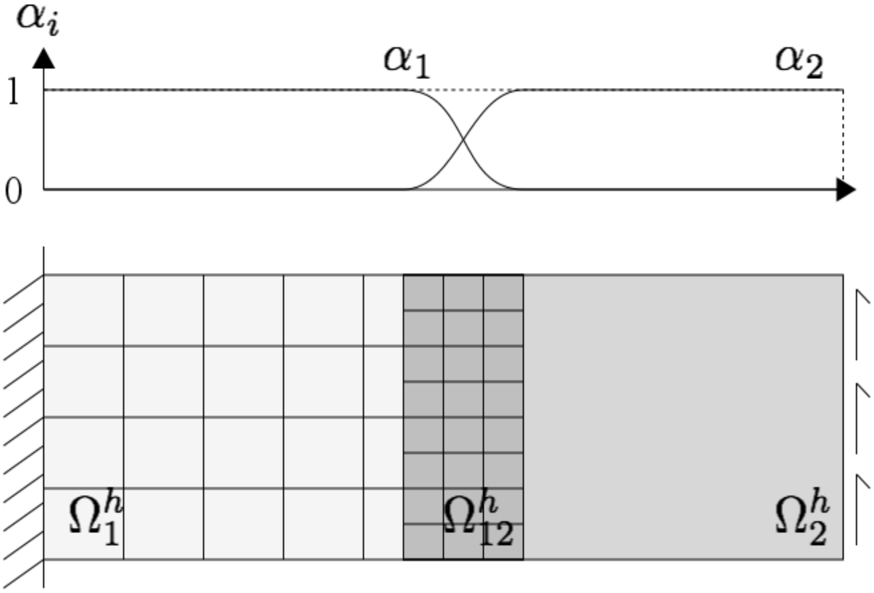


Figure 1.8: Partially overlapping application of the Arlequin method.

The Lagrange multipliers $\boldsymbol{\lambda}$ are applied to the overlapped zone $\Omega_1 \cap \Omega_2$ and are modeled with the mesh Ω_{12}^h shown in Figure 1.9, in which the explosion of the overlapped models of Figure 1.8 is illustrated. Note that the displacements fields $\mathbf{u}_1$ and $\mathbf{u}_2$ are modeled with different meshes, described in Figure 1.9 as Ω_1^h and Ω_2^h, respectively.

A coupling between implicit dynamic time integration schemes is performed in [Ghanem et al., 2013]. In that paper, a displacement continuity is enforced, the choice to impose same weight functions (e.g. as in Figure 1.8) for the different energy terms is assumed, so that $\alpha_i = \eta_i = \varphi_i$, and three kinds of continuity operator $C(\square,\square)$ are considered:

- the so-called "L^2 coupling":

$$C(\boldsymbol{\lambda}, \mathbf{u}_1 - \mathbf{u}_2) = \int_{\Omega_{12}} \boldsymbol{\lambda} \cdot (\mathbf{u}_1 - \mathbf{u}_2)\, d\Omega \tag{1.80}$$

that leads to an ill-conditioned global stiffness matrix depending on the application;

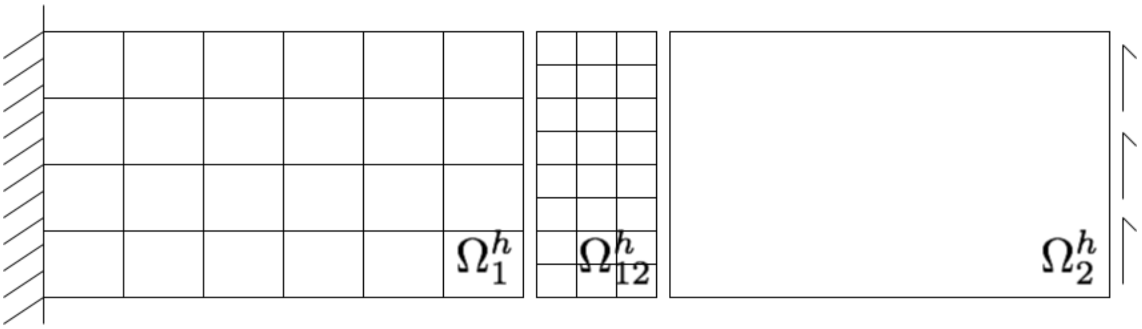


Figure 1.9: Meshes for the partially overlapping Arlequin method.

- the so-called "H^1 coupling":

$$C(\boldsymbol{\lambda}, \mathbf{u}_1 - \mathbf{u}_2) = \int_{\Omega_{12}} \left(\boldsymbol{\lambda} \cdot (\mathbf{u}_1 - \mathbf{u}_2) + l^2 \varepsilon(\boldsymbol{\lambda}) : \varepsilon(\mathbf{u}_1 - \mathbf{u}_2)\right) \, d\Omega \tag{1.81}$$

  with the introduction of a parameter l homogeneous to a length;

- the so-called "energy-based coupling":

$$C(\boldsymbol{\lambda}, \mathbf{u}_1 - \mathbf{u}_2) = \int_{\Omega_{12}} \sigma(\boldsymbol{\lambda}) : \varepsilon(\mathbf{u}_1 - \mathbf{u}_2) \, d\Omega \tag{1.82}$$

  that is the choice for the Arlequin application in [Ghanem et al., 2013].

The numerical integration of the weak formulation leads to a system similar to the non-overlapping Domain Decomposition one. Here, the non-conform meshes are directly dealt with in the connectivity matrices $\mathbf{C}_i$, that are not unitary Boolean matrices but full rectangular matrices, and their numerical integration requires the employment of specific quadrature techniques because of the differences between the overlapped meshes.

For wave propagation problems, in contrast to non-uniform meshes and non-overlapping Domain Decomposition methods, the Arlequin method offers the possibility to attenuate or avoid the wave-reflection due to the different discretizations of the subdomains.

4.3.1 Arlequin framework for dynamic adaptivity

As the non-overlapping Domain Decomposition methods, the partially-overlapping Arlequin method should be coupled with a re-meshing strategy to attain the industrial goal. Developments of the completely-overlapping Arlequin method could be interesting as global-local couplings because of the possibility to leave unchanged both global and local meshes applied everywhere, playing with the parameters α_i in order to activate or deactivate the model refinement. Nevertheless, the requirement of a detailed fine mesh everywhere could be costly in terms of memory allocation and the complexity of connectivity matrices $\mathbf{C}_i$ and Lagrange multipliers $\boldsymbol{\lambda}$ definitions could be costly in terms of computing time.

4.4 Global-local approaches and Multi-grid algorithms

The global-local approaches constitute an alternative to the aforementioned coupling strategies and can be classified as completely overlapping Domain Decomposition methods. The idea to couple a global linear analysis with given global mesh and a localized analysis with a refined mesh and the eventual presence of defects was introduced in [Whitcomb, 1991]. The iterations required by the process were shown to be more advantageous rather than a monolithic nonlinear analysis with a non-uniform mesh. Furthermore, keeping the global analysis with an unchanged global model is said to be "non-intrusive" for couplings, avoiding re-meshing or re-formulations during the analysis.

The same philosophy has been followed in the works for coupling research and industrial codes in [Gendre et al., 2009], [Gendre et al., 2011] and [Guguin et al., 2014]. In those cases, based on a global-local superposition technique and for static problems, the purpose was to couple in a non-intrusive manner a commercial code applied to the overall problem domain as global analysis, with a research code applied only to the zones of interest with a detailed model. The numerical issues connected to the iterative procedures and to the stress mapping represented additional research challenges of those works.

The challenges of the present works are deeply different. Transient dynamics solved with explicit time integration solvers is prone to numerical and operational issues differing from static problems and the same intrusivity characteristic denotes different issues.

Another class of completely overlapping coupling methods is represented by the Multi-grid algorithms. Introduced in [Brandt, 1977], [McCormick and Ruge, 1982] and [Parsons and Hall, 1990], they were employed to enhance the accuracy of the standard iterative solvers for highly irregular solutions. Applications of Finite Element discretizations to elliptic problems are also presented in [Fish and Belsky, 1995], [Fish et al., 1996] and [Fish et al., 1997] for stratified composites or in [Glowinski et al., 2005] with a mathematical study of arbitrary local patch discretizations. An application to crack propagation in statics has been presented in [Passieux et al., 2013]. Once again, the global analysis is carried out in the overall structure by means of a linear elastic analysis (potentially with a commercial software) and a local patch is activated with the use of nonlinear crack predicting technologies. In that case, a Multi-grid solver has been employed for bridging the global and local analyses.

5 Existing coupling techniques inside Abaqus

5.1 Impact test example

The impact test case performed in the MAAXIMUS report [Heimbs, 2011] and described in Figure 1.10 is taken as example for the practical description of the existing coupling techniques inside Abaqus. A flat panel made of a stratified composite is impacted by a glass ball.

The interest of the example resides in the possibility to refine the model around the impacted zone of the panel, keeping a coarse model in the surroundings in order to save

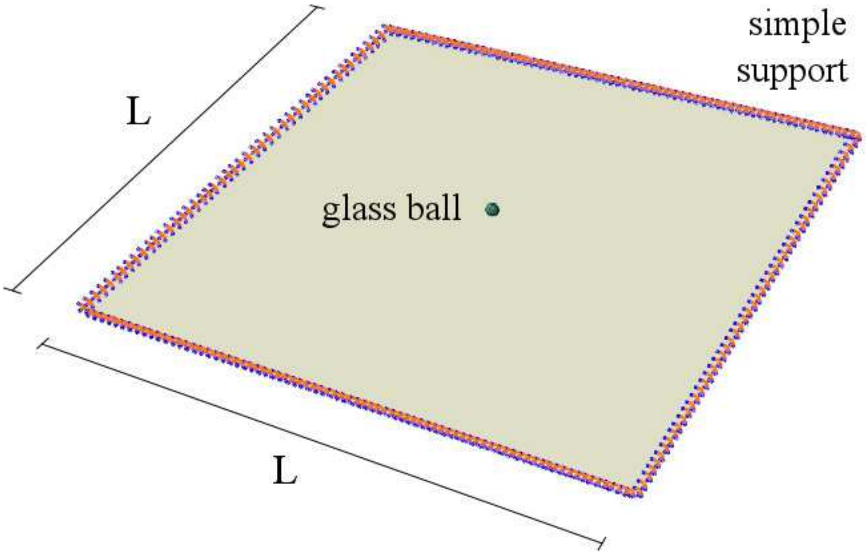


Figure 1.10: Flat panel in stratified composite impacted by a glass ball.

computing time and to maintain acceptable the levels of accuracy.

The problem data are as follows:

- for the glass impactor ball:
  - mass of 18.6 g;
  - diameter of 24.4 mm;
  - impact velocity of 100 m/s;
- for the composite panel:
  - size of the panel $L = 1\ m$;
  - thickness of 1.5 mm;
  - 12 unidirectional plies governed by a damage law.

The three main coupling techniques integrated in the Abaqus code v6.10 have been studied: the Submodeling, the Subcycling and the Co-Simulation techniques.

5.2 Submodeling technique

Submodeling is the technique of studying a local part of a model with a refined mesh, based on the interpolation of the solution from an initial global model onto appropriate parts of the boundary of the *submodel*. The method is useful when it is necessary to obtain an accurate, detailed solution in the local region and the fine discretization of that local region has negligible effect on the overall solution. The response at the boundary of a local region is defined by the solution for the global model and it determines the solution in the *submodel*, together with any loads applied to the local region. The technique relies on the global model defining this *submodel* boundary response with sufficient accuracy.

With a few restrictions, different element types can be used in the *submodel* compared to those used to model the corresponding region in the global model. The material response defined for the *submodel* may even be different from that defined for the global model. Both the global model and the *submodel* can have nonlinear responses.

A Submodeling analysis consists of:

- running a global analysis and saving the results in the vicinity of the *submodel* boundary;
- defining the total set of driven nodes or driven surfaces in the *submodel*;
- defining the time variation of the driven variables in the *submodel* analysis by specifying the actual nodes and degrees of freedom or element-based surfaces to be driven in each step;
- running the *submodel* analysis using the driven variables to drive the solution.

The *submodel* is run as a separate analysis. The only link between the *submodel* and the global model is the transfer of the time-dependent values of the variables to the relevant driven variables of the *submodel.*

To summarize, the Submodeling technique can be generally used:

- to study a local part of a model with a refined mesh based on interpolation of the solution from an initial, relatively coarse, global model;
- to obtain an accurate, detailed solution in a local region when the detailed modeling of that local region has negligible effect on the overall solution;
- to drive a local part of the model by nodal results, such as displacements, or by the element stress results from the global mesh;
- to analyze an acoustic model driven by displacements from a structural, global model when the acoustic fluid has negligible effect on the structural solution;
- for the analysis of a structure driven by acoustic pressures from an acoustic, or coupled acoustic-structural, global model;
- a combination of explicit and implicit procedures;
- a combination of linear and nonlinear procedures.

Figure 1.11 represents the application of the Submodeling technique. At first, a coarse global analysis is carried out in the overall structure. Then, a finer local analysis is applied to the impacted zone, extrapolating from the global solution the displacement boundary conditions along the red lines. In this way, the effects of the finer solution to the global response are not considered.

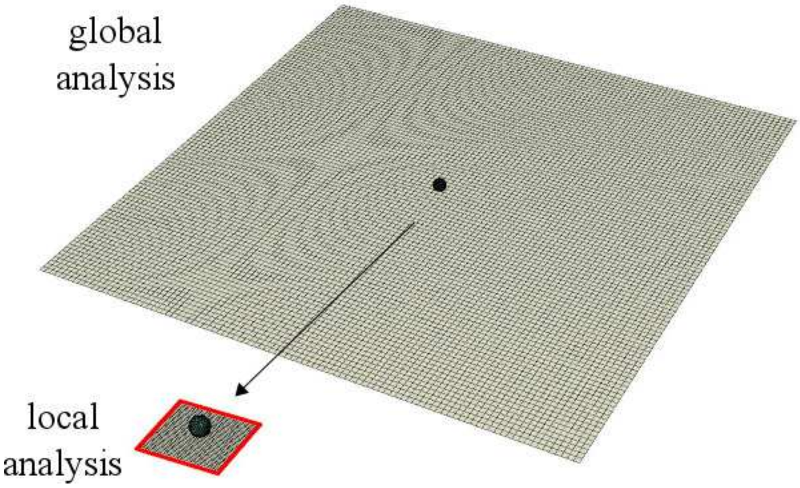


Figure 1.11: Submodeling technique.

Algorithm 3 summarizes the steps of the sequential Submodeling approach.

```
while $n < N$ do
    global solution;
    update $n$;
end
mapping of global solution history;
definition of local boundary conditions history $\leftarrow$ global solution;
while $m < N \cdot M$ do
    local solution;
    update $m$;
end
```

Algorithm 3: Submodeling coupling scheme.

A two-way coupling between the global and local analyses should be preferred. Nevertheless, in contrast to the Domain Decomposition via Co-Simulation, non-conforming meshes can be employed with Submodeling exploiting the mapping of the global solution into the local mesh.

5.3 Subcycling technique

The Subcycling technique in Abaqus/Explicit is based on the nodal partitioning originally introduced in [Belytschko et al., 1979] and described in Section 4.1.2 for the application of different time steps to the zones of a non-uniform mesh. The Subcycling zones remain unchanged during the analysis and each Subcycling zone is integrated with its own stable time step. The subcycle time step sizes are chosen as integer divisors of the time step used in the *master* domain. Therefore, all the domains exactly reach the same time points of the *master* domain.

An arbitrary number of Subcycling zones can be defined. However, some modeling features cannot be split between Subcycling zones. In that case, Abaqus/Explicit automatically merges the Subcycling zones. For instance, Subcycling zones are merged when general contact is specified over a surface shared by the zones. Then, when Subcycling zones are merged, the smallest stable time step among the merged zones is used. The constraint, connector or rigid body can be assigned to the Subcycling zone only if every node belongs to that Subcycling zone.

Efficient Subcycling requires a proper choice of Subcycling zones. For each Subcycling zone, the time step size should be small compared to the *master* zone, producing a large number of subcycles, i.e. the ratio of the stable time step size in the *master* zone to the stable time step sizes in the Subcycling zones. In addition to a large number of subcycles, the number of elements in a Subcycling zone should generally be small compared to the total number of elements in the model for optimal performance benefit. If a majority of elements in the model are in Subcycling zones, there will not be much performance benefit.

The Subcycling algorithm used in Abaqus/Explicit provides sufficient accuracy for most complex dynamic models. However, because of the interpolation used on interface nodes, Subcycling solutions can introduce a truncation error. This error should not affect the overall dynamic behavior of the model. Special attention should be given to the interfaces when general contact is involved: it is not necessary to define a pair of surfaces that have the potential for contacting each other within the same zone; however, to minimize truncation errors, it is highly recommended that a single surface that has the potential for contacting others is not split across the zones.

Figure 1.12 represents the application of a non-uniform mesh, finer in the impacted zone. The Subcycling technique is adopted to use two different time steps in the two partitions. The zone evidenced in red (*subcycling* zone) is the node partition (defined as *element set*) in which the time step is smaller than in the rest of the structure (*master* zone).

As described in [Daniel, 2003], the Subcycling algorithm simplified into two partitions, *master* and *subcycling*, is summarized in Algorithm 4.

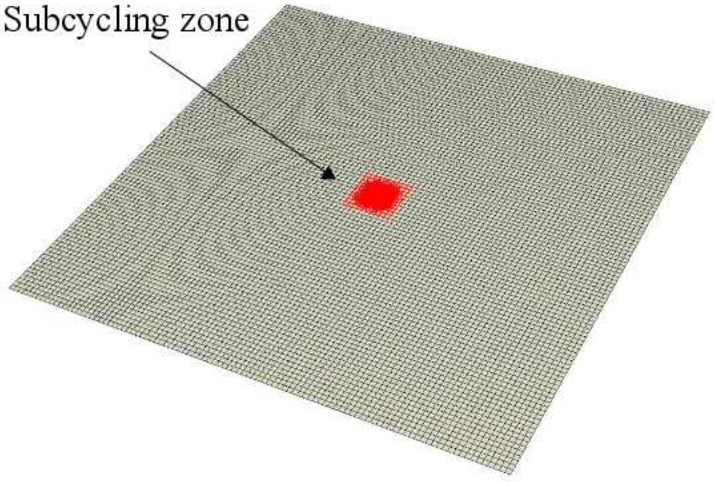


Figure 1.12: Non-uniform mesh and Subcycling technique.

```
while $n \leq N$ do
    momentum balance at the master nodes;
    update velocities $^{n+\frac{1}{2}}\mathbf{V}^{\text{master}}$;
    while $m \leq M$ do
        momentum balance at the subcycling nodes;
        update mid-step velocities $^{m+\frac{1}{2}}\mathbf{V}^{\text{subcycling}}$;
        update $m$;
    end
    update interface displacements averaging the velocities $^{n+\frac{1}{2}}\mathbf{V}^{\text{master}}$ and $^{m+\frac{1}{2}}\mathbf{V}^{\text{subcycling}}$;
end
```

Algorithm 4: Subcycling scheme.

5.4 Co-Simulation technique

The Co-Simulation technique is a capability for run-time coupling of Abaqus and other analysis programs: an Abaqus analysis can be coupled to another Abaqus analysis or to a third-party analysis program to perform multi-physics or multi-model simulations.

Co-Simulation between Abaqus/Standard (implicit solver) and Abaqus/Explicit (explicit solver) illustrates a multiple domain analysis approach using the algorithm introduced in [Gravouil and Combescure, 2001], originally developed for arbitrary time integration schemes couplings and specified in Section 4.2.3 as explicit-explicit coupling application. Each analysis operates on a complementary zone of the domain where it is expected to provide the more computationally efficient solution. For example, the implicit solver provides a more efficient solution for light and stiff components, while the explicit solver is more efficient for solving complex contact interactions.

The Co-Simulation technique can more generally be used to solve:

- fluid-structure interactions by coupling Abaqus (/Explicit or /Standard) with Computational Fluid Dynamic (CFD) analysis programs;
- conjugate heat transfer problems by coupling Abaqus/Standard with CFD analysis programs;
- electromagnetic-thermal or electromagnetic-mechanical interactions by coupling Abaqus with electromagnetic analysis programs;
- multi-physics simulations by coupling Abaqus (/Explicit or /Standard) with third-party analysis programs;
- Domain Decomposition analyses by coupling Abaqus/Standard to /Explicit.

In a Co-Simulation, the interaction between the domains goes through a common physical interface region over which data are exchanged in a synchronized manner between Abaqus and the coupled analysis program.

One domain may affect the response of another domain through one or more of the following:

- constitutive behavior, such as the yield stress defined as a function of temperature or stress defined as a function of other solution fields, such as thermal strains or the piezoelectric effect;
- surface tractions/fluxes, such as a fluid exerting pressure on a structure;
- body forces/fluxes, such as heat generation due to flow of current in a coupled thermal-electrical simulation;
- contact forces, such as the forces due to contact between separate domains;
- kinematics, such as fluid in contact with a compliant structure where the interface motion affects the fluid flow.

The coupling between Abaqus and another analysis program is carried out using the Co-Simulation Engine. In particular, the interaction between the implicit and explicit models takes place through a common interface region. When the implicit and explicit meshes differ, the following limitations apply:

- in cases where the stress state near the Co-Simulation interface is significant (approaching 1% or more) relative to the material stiffness, an appreciable irregular mesh distortion may be observed if the mesh density adjacent to the Co-Simulation region differs greatly between the explicit and implicit models;

- a Co-Simulation region node cannot be a slave node in a tie constraint or a kinematic coupling constraint.

Stability and accuracy of the Co-Simulation solution may be adversely affected when the following model features are defined at or near the Co-Simulation region:

- connector elements connected to Co-Simulation region nodes;
- Co-Simulation region nodes that participate in a tie constraint or a kinematic coupling constraint.

Figure 1.13 represents the application of the Domain Decomposition into two subdomains, a finer subdomain in the impacted zone using the explicit solver, and a coarse one consisting of the rest of the structure with the implicit solver. The Co-Simulation technique connects the two subdomains through the interfacial nodes evidenced in red. The finer mesh in the explicit subdomain is generated in such a way that the meshes are conforming through the interface, because the nodes in the Co-Simulation regions cannot be slave nodes in a tie constraint.

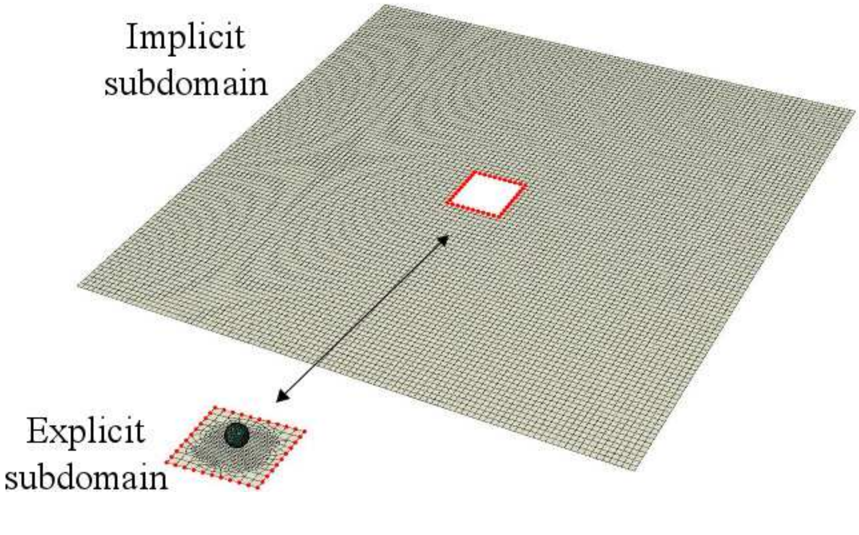


Figure 1.13: Co-Simulation technique.

The *GC* algorithm introduced in [Gravouil and Combescure, 2001] is integrated in Abaqus following the Algorithm 5. The Lagrange multipliers are computed in the /Standard environment at each fine time step because it requires a solver for linear systems, not available in /Explicit. So, at each fine time step, velocities and masses are sent to /Standard and the Lagrange multipliers are sent to /Explicit.

```
while $n < N$ do
    /Standard resolution: free momentum balance;
    while $m < M$ do
        /Explicit resolution: free momentum balance;
        /Explicit sends to /Standard: mass and velocity;
        /Standard resolution: Lagrange multipliers;
        /Standard sends to /Explicit: Lagrange multipliers;
        /Explicit resolution: link correction with Lagrange multipliers;
        update $m$;
    end
    /Standard resolution: link correction with Lagrange multipliers;
    update $n$;
end
```

Algorithm 5: Co-Simulation /Standard-/Explicit coupling scheme.

5.5 Comparison between the techniques in [Heimbs, 2011]

Different models have been tested and coupled together:

1. non-uniform meshes using:
    (a) a *macro*-scale: one layered shell element over the thickness of the panel, or
    (b) a *pseudo-meso*-scale: three layered shell elements and two cohesive interfaces over the thickness of the panel;
2. coupling the two scales as described in Figure 1.14.

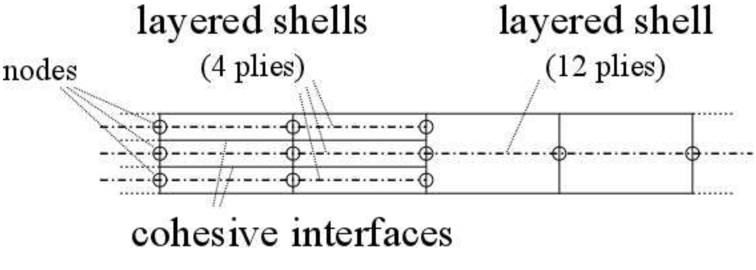


Figure 1.14: Coupling *macro*- and *pseudo-meso*-scales.

Table 1.5 summarizes some remarks related to the Subcycling, Co-Simulation and Submodeling techniques in the MAAXIMUS report [Heimbs, 2011]. When coupling together two *macro*-scale meshes, the Subcycling technique is the most advantageous coupling technique in terms of computing cost and accuracy, slightly more convenient than the monolithic computation with non-uniform mesh and unique time step. An analogous restricted gain is obtained with the *pseudo-meso*-scale meshes. On the other hand, coupling the *macro*-scale with the *pseudo-meso*-scale is remarked as a very interesting strategy when run with a unique time step, whereas the Subcycling presents some issues, both in computing time and in accuracy.

model	*discretization and technique*	*CPU time*	*accuracy*
macro (reference 1)	uniform fine mesh	100%	
	non-uniform mesh, unique time step	9.81 %	good
	non-uniform mesh, **Subcycling**	8.11 %	good
	compatible meshes, **Co-Simulation**	74.24 %	good
	non-conforming meshes, **Submodeling**	11.17 %	bad
pseudo-meso (reference 2)	uniform fine mesh	100%	
	non-uniform mesh, unique time step	15.87 %	good
	non-uniform mesh, **Subcycling**	15.64 %	good
coupling	unique time step	4.17 %	good
	Subcycling	5.07 %	bad

Table 1.5: MAAXIMUS-WP3.5 report [Heimbs, 2011], Abaqus v6.10.

Note that the Co-Simulation technique does not offer appreciable results in terms of computing time, probably because of the employment of the implicit time integration scheme over an extended surface of the structure. The application of an explicit-explicit coupling scheme should theoretically give performances similar to the Subcycling technique.

6 Conclusions

Starting from a genetical initial-boundary value problem, the most common Finite Element time integration schemes for dealing with impact problems have been presented, together with advanced multi-scale coupling techniques.

Special attention has been given to the existing capabilities and coupling strategies within the commercial software Abaqus.

In the studies, the non-overlapping Domain Decomposition method for multi-scale coupling developed in [Gravouil and Combescure, 2001] and further works, applied to Central Difference schemes for the time integration, has risen as a robust and efficient technique, suitable for the industrial goal if coupled with a re-meshing strategy. Due to the interest of such approach, it has been implemented and studied, as described in the what follows.

Chapter 2

Investigation and implementation of the Finite Element Tearing and Interconnecting (FETI) approach

In this Chapter, the implementation of the non-overlapping Domain Decomposition algorithms is described. The application to explicit-explicit coupling is reported and implemented. This investigation aims at studying the evolution from the GC to the GCbis algorithms focusing on the stability characteristic.

Contents

1 Introduction

The non-overlapping Domain Decomposition methods emerged in Chapter 1 as robust and efficient multi-scale coupling technique, even if requiring a re-meshing strategy to optimize the utilization of finer modeling in case of evolutive phenomena. A deeper study of the technical aspects of Domain Decomposition method has been so considered, notably because of difficulties first encountered when trying to develop the Substitution method.

In particular, the multi-time-step FETI approaches developed in [Gravouil and Combescure, 2001] and the energy-preserving version proposed in [Prakash and Hjelmstad, 2004] are here presented for explicit-explicit couplings, with details about the algorithms, resulting comparisons and remarks. Such methods will be used as reference for the developments in the thesis.

2 Multi-time-step FETI approach

2.1 Formulation

Separating the solution in *free* and *link* terms, the System (1.57) is developed with five sets of unknowns ${}^{m+1}\mathbf{A}_1^{\text{free}}$, ${}^{m+1}\mathbf{A}_1^{\text{link}}$, ${}^{n+1}\mathbf{A}_2^{\text{free}}$, ${}^{n+1}\mathbf{A}_2^{\text{link}}$ and ${}^{m+1}\mathbf{\Lambda}$ as follows $\forall n \in \{0,\dots,N-1\}$:

$$\begin{array}{ll}
\mathbf{M}_1 {}^{m+1}\mathbf{A}_1^{\text{free}} = {}^{m+1}\mathbf{F}_1^{\text{ext}} - {}^{m+1}\mathbf{F}_1^{\text{int}} & \text{in } \Omega_1^h|_{t^{m+1}} \ \forall m \in \{0,\dots,M-1\} \\
\mathbf{M}_{1,\Gamma} {}^{m+1}\mathbf{A}_{1,\Gamma}^{\text{link}} = -\, {}^{m+1}\mathbf{\Lambda} & \text{over } \Gamma_1^h|_{t^{m+1}} \ \forall m \in \{0,\dots,M-1\} \\
\mathbf{M}_2 {}^{n+1}\mathbf{A}_2^{\text{free}} = {}^{n+1}\mathbf{F}_2^{\text{ext}} - {}^{n+1}\mathbf{F}_2^{\text{int}} & \text{in } \Omega_2^h|_{t^{n+1}} \\
\mathbf{M}_{2,\Gamma} {}^{n+1}\mathbf{A}_{2,\Gamma}^{\text{link}} = \Pi_h^{1T}\, {}^{n+1}\mathbf{\Lambda} & \text{over } \Gamma_2^h|_{t^{n+1}} \\
{}^{m+1}\mathbf{V}_{1,\Gamma} = \Pi_h^1\, {}^{m+1}\mathbf{V}_{2,\Gamma} & \text{over } \Gamma_1^h|_{t^{m+1}} \ \forall m \in \{0,\dots,M-1\}
\end{array} \tag{2.1}$$

where, with reference to Section 4.2 of Chapter 1, Ω_1^h, Ω_2^h and Γ_1^h are defined in Figure 1.6 and the time domain $[t_{\text{in}}, t_{\text{end}}]$ is differently discretized for the two subdomains as described in Figure 1.7.

The resolution strategy of System (2.1) derives from the development of the last equation using the Central Difference scheme and the *free-link* variable separation in Equation (1.61):

$$\begin{aligned}
&{}^{m+\frac{1}{2}}\mathbf{V}_{1,\Gamma} + \left({}^{m+1}\mathbf{A}_{1,\Gamma}^{\text{free}} + {}^{m+1}\mathbf{A}_{1,\Gamma}^{\text{link}}\right)\frac{\Delta t_1}{2} = \\
&\qquad = \Pi_h^1\, {}^{m+\frac{1}{2}}\mathbf{V}_{2,\Gamma} + \Pi_h^1 \left({}^{m+1}\mathbf{A}_{2,\Gamma}^{\text{free}} + {}^{m+1}\mathbf{A}_{2,\Gamma}^{\text{link}}\right)\frac{\Delta t_2}{2}.
\end{aligned} \tag{2.2}$$

Then, using the second and the fourth equations in System (2.1), Equation (2.2) is expressed in function of Lagrange multipliers ${}^{m+1}\boldsymbol{\Lambda}$ and becomes:

$$\begin{aligned} &{}^{m+\frac{1}{2}}\mathbf{V}_{1,\Gamma} + \left({}^{m+1}\mathbf{A}^{\text{free}}_{1,\Gamma} - \mathbf{M}^{-1}_{1,\Gamma}\,{}^{m+1}\boldsymbol{\Lambda}\right)\frac{\Delta t_1}{2} = \\ &= \Pi^1_h\,{}^{m+\frac{1}{2}}\mathbf{V}_{2,\Gamma} + \Pi^1_h\left({}^{m+1}\mathbf{A}^{\text{free}}_{2,\Gamma} + \mathbf{M}^{-1}_{2,\Gamma}\Pi^{1T}_h\,{}^{m+1}\boldsymbol{\Lambda}\right)\frac{\Delta t_2}{2} \end{aligned} \tag{2.3}$$

that, switching the positions in order to compute the Lagrange multipliers ${}^{m+1}\boldsymbol{\Lambda}$:

$$\begin{aligned} \left(\frac{\Delta t_1}{2}\mathbf{M}^{-1}_{1,\Gamma} + \frac{\Delta t_2}{2}\Pi^1_h\mathbf{M}^{-1}_{2,\Gamma}\Pi^{1T}_h\right){}^{m+1}\boldsymbol{\Lambda} &= {}^{m+\frac{1}{2}}\mathbf{V}_{1,\Gamma} + {}^{m+1}\mathbf{A}^{\text{free}}_{1,\Gamma}\frac{\Delta t_1}{2} + \\ &- \Pi^1_h\,{}^{m+\frac{1}{2}}\mathbf{V}_{2,\Gamma} - \Pi^1_h\,{}^{m+1}\mathbf{A}^{\text{free}}_{2,\Gamma}\frac{\Delta t_2}{2} \end{aligned} \tag{2.4}$$

where the coefficients matrix is the so-called *condensation* matrix $\mathbb{H}$:

$$\mathbb{H} \triangleq \frac{\Delta t_1}{2}\mathbf{M}^{-1}_{1,\Gamma} + \frac{\Delta t_2}{2}\Pi^1_h\mathbf{M}^{-1}_{2,\Gamma}\Pi^{1T}_h \tag{2.5}$$

and, for the notation in the followings:

$${}^{m+1}\mathbf{V}^{\text{free}}_{i,\Gamma} \triangleq {}^{m+\frac{1}{2}}\mathbf{V}_{i,\Gamma} + {}^{m+1}\mathbf{A}^{\text{free}}_{i,\Gamma}\frac{\Delta t_i}{2}. \tag{2.6}$$

So, System (2.1) is solved for every coarse time step Δt_2 in five steps:

$$\begin{array}{ll} \text{1. coarse free solution} & \mathbf{M}_2{}^{n+1}\mathbf{A}^{\text{free}}_2 = {}^{n+1}\mathbf{F}^{\text{ext}}_2 - {}^{n+1}\mathbf{F}^{\text{int}}_2 \\ \text{2. fine free solution} & \mathbf{M}_1{}^{m+1}\mathbf{A}^{\text{free}}_1 = {}^{m+1}\mathbf{F}^{\text{ext}}_1 - {}^{m+1}\mathbf{F}^{\text{int}}_1 \\ \text{3. interface constraint} & \mathbb{H}\,{}^{m+1}\boldsymbol{\Lambda} = {}^{m+1}\mathbf{V}^{\text{free}}_{1,\Gamma} - \Pi^1_h\,{}^{m+1}\mathbf{V}^{\text{free}}_{2,\Gamma} \\ \text{4. fine link solution} & \mathbf{M}_{1,\Gamma}{}^{m+1}\mathbf{A}^{\text{link}}_{1,\Gamma} = -\,{}^{m+1}\boldsymbol{\Lambda} \\ \text{5. coarse link solution} & \mathbf{M}_{2,\Gamma}{}^{n+1}\mathbf{A}^{\text{link}}_{2,\Gamma} = \Pi^{1T}_h\,{}^{n+1}\boldsymbol{\Lambda} \end{array} \tag{2.7}$$

where steps 2, 3 and 4 are re-iterated for $m = 0, \ldots, M-1$.

Remark 6 *The condensation matrix* $\mathbb{H}$ *is a square non-diagonal matrix, which size is* $\dim\mathbb{H} = \left(\mathcal{N}\dim\Gamma^h_1\right)$. *The extra-diagonal terms come from the down-scaling operation* $\Pi^1_h\square\Pi^{1T}_h$ *of the inverted mass matrix of the coarse subdomain through a linear interpolation in space. So, in contrast to the acceleration resolutions, the computation of Lagrange multipliers requires a solver of linear systems.*

The steps of System (2.7) are summarized in Algorithm 6.

```
computation of $\mathbb{H}$;
while $t_2 < t_{end}$ do
    coarse free solution: $\mathbf{M}_2{}^{n+1}\mathbf{A}_2^{\text{free}} = {}^{n+1}\mathbf{F}_2^{\text{ext}} - {}^{n+1}\mathbf{F}_2^{\text{int}}$;
    while $t_1 < t_2 + \Delta t_2$ do
        fine free solution: $\mathbf{M}_1{}^{m+1}\mathbf{A}_1^{\text{free}} = {}^{m+1}\mathbf{F}_1^{\text{ext}} - {}^{m+1}\mathbf{F}_1^{\text{int}}$;
        interface constraint: $\mathbb{H}\;{}^{m+1}\mathbf{\Lambda} = {}^{m+1}\mathbf{V}_{1,\Gamma}^{\text{free}} - \Pi_h^1\;{}^{m+1}\mathbf{V}_{2,\Gamma}^{\text{free}}$;
        fine link solution: $\mathbf{M}_{1,\Gamma}{}^{m+1}\mathbf{A}_{1,\Gamma}^{\text{link}} = -\;{}^{m+1}\mathbf{\Lambda}$;
        update $t_1 \leftarrow t_1 + \Delta t_1$;
    end
    coarse link solution: $\mathbf{M}_{2,\Gamma}{}^{n+1}\mathbf{A}_{2,\Gamma}^{\text{link}} = \Pi_h^{1T}\;{}^{n+1}\mathbf{\Lambda}$;
    update $t_2 \leftarrow t_2 + \Delta t_2$;
end
```

Algorithm 6: GC algorithm.

2.2 Application to assess the coupling properties

The multi-time-step FETI approach is so applied with the *GC* algorithm in order to analyze the coupling properties due to the interface constraint. The overall domain of the problem is split in two subdomains as described in Figure 2.1, where the boundary conditions, loads and sizes are recalled.

The problem data are:

- density $\rho = 7800\ kg/m^3$;
- Young's modulus $E = 210 \cdot 10^9\ Pa$;
- Poisson's ratio $\nu = 0.3$;
- length $L = 30\ m$;
- height $H = 10\ m$;
- uniform shear load $\overline{f} = 3 \cdot 10^6\ Pa$ constant in time.

The numerical solution is monitored at four representative points, at the locations shown in Figure 2.1:

- the location A inside the first subdomain Ω_1 in order to observe the fine solution,
- the locations B_1 and B_2 over the interface Γ at a corresponding position in subdomains Ω_1 and Ω_2, respectively, in order to exhibit the transmission properties that characterize the given coupling scheme,
- the location C at the extremity of the structure in the second subdomain Ω_2 in order to observe the global response of the structure.

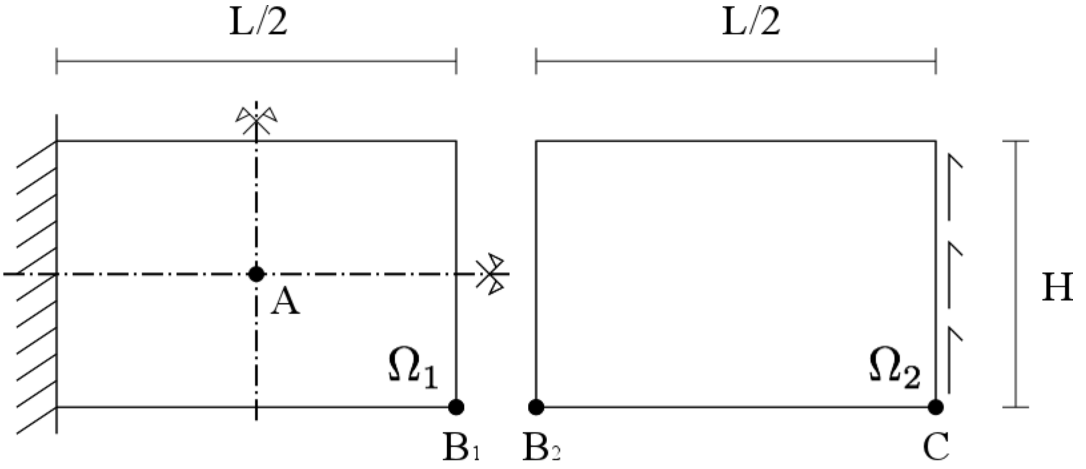
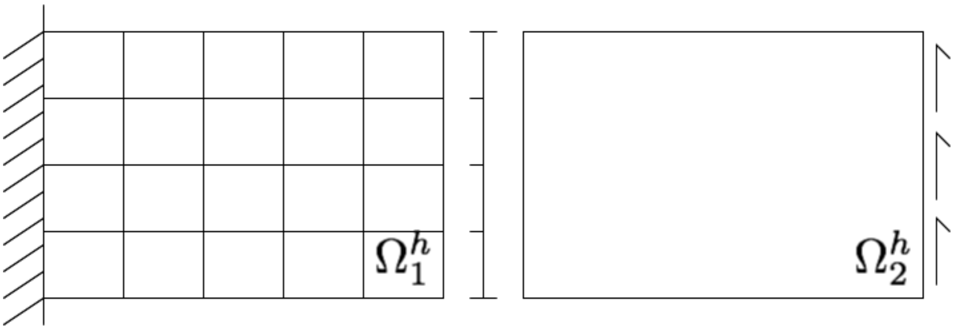
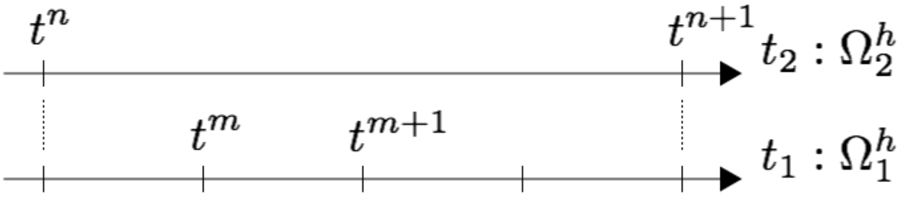


Figure 2.1: Boundary conditions, loads, sizes and monitored locations A, B_1, B_2 and C.

Figure 2.2 shows the spatial and temporal discretizations applied to the two subdomains. The fine subdomain Ω_1^h is discretized with 20 displacement-based 4-node quadrilateral elements and an associated time step $\Delta t_1 = 3 \cdot 10^{-4}$ s, whereas the coarse subdomain Ω_2^h is discretized with only 1 quadrilateral element of the same type and an associated time step $\Delta t_2 = 12 \cdot 10^{-4}$ s. In this case, the subdomains mesh ratio $h_{2/1}$, defined as the ratio of the coarse element length to the fine, corresponds to the time step ratio M, defined as the ratio of the coarse time step to the fine one, so that $h_{2/1} = M = 4$.



(a) spatial discretization with $h_{2/1} = 4$



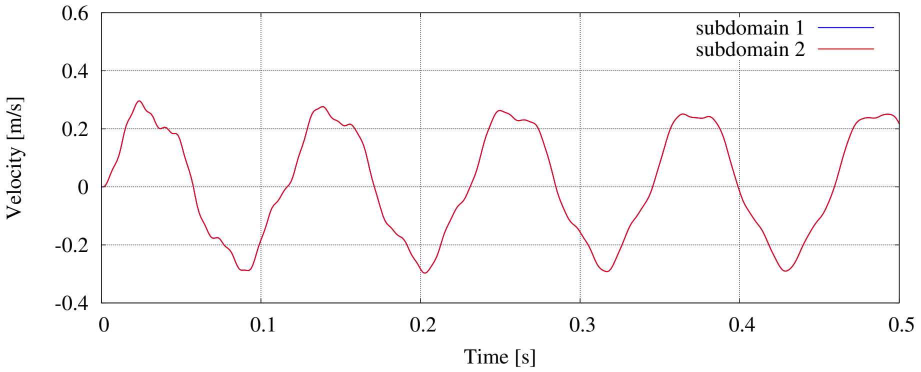
(b) temporal discretization with $M = 4$

Figure 2.2: Discretization for non-overlapping Domain Decomposition.

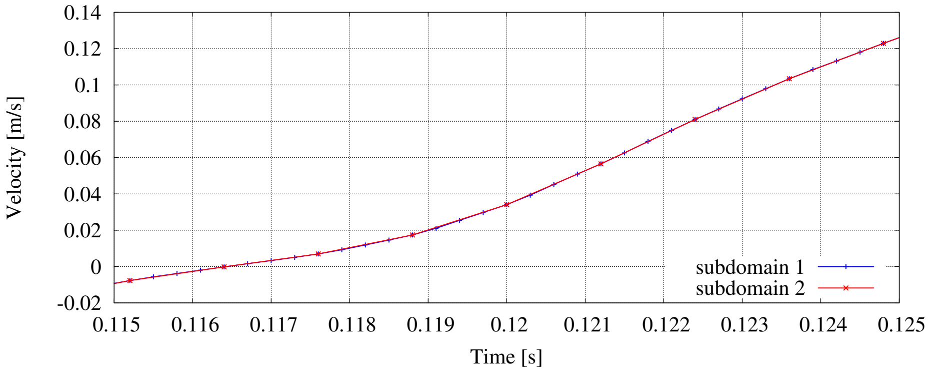
Figure 2.3 shows the vertical velocities at the locations B_1 (subdomain 1) and B_2

(subdomain 2). According to the interface constraint, the two curves are overlapped in Figure 2.3(a) and in the shorter interval of Figure 2.3(b). On the contrary, small discrepancies between the curves are visible in Figure 2.3(c): the continuity is exact at the coarse time instants, whereas a small gap is produced at the beginning of the fine time stepping and shrinks to the following coarse time instant. Indeed, a perfect continuity over the fine time steps is not compatible with the Central Difference scheme, because linear velocities in the fine time stepping imply constant accelerations in the coarse time steps with jumps at the coarse time instants, in contrast with continuous time integration schemes.

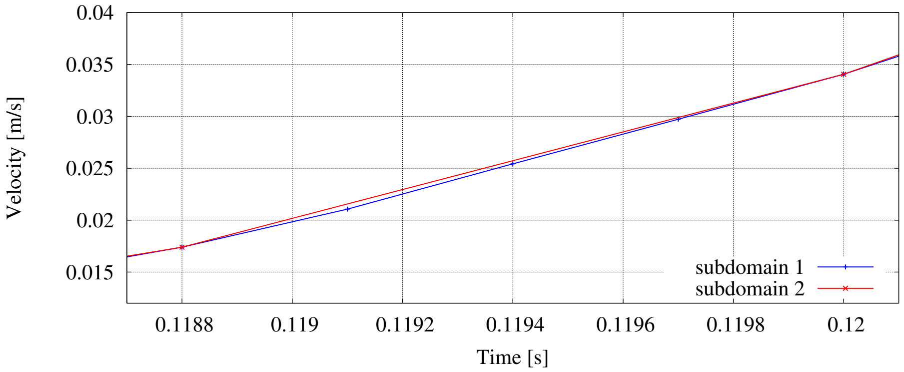
Figure 2.4 shows the vertical accelerations at the locations B_1 (subdomain 1) and B_2 (subdomain 2). As expected, the accelerations are not continuous along the interface. In addition, Figure 2.4(b) highlights the shape of the accelerations in the fine subdomain (blue line) that is source of discrepancy between the velocities in the fine time stepping.



(a) between $t_{\text{in}} = 0\ s$ and $t_{\text{end}} = 0.5\ s$

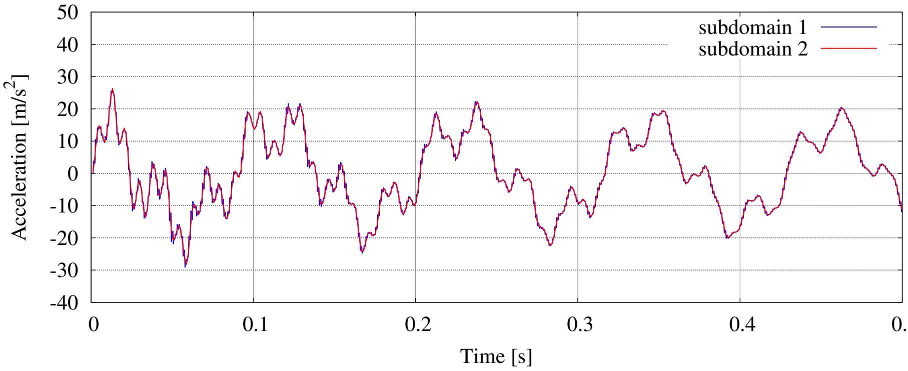


(b) between $t_1^{\text{zoom}} = 0.115\ s$ and $t_2^{\text{zoom}} = 0.125\ s$

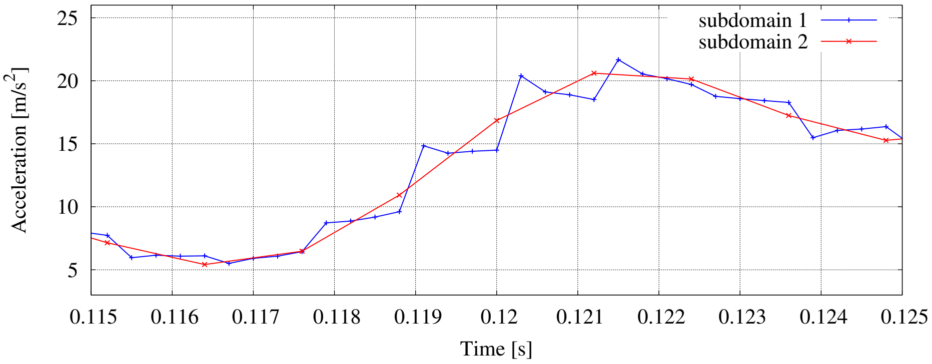


(c) between $t_1^{\text{zoom'}} = 0.1187\ s$ and $t_2^{\text{zoom'}} = 0.1201\ s$

Figure 2.3: Vertical velocities at the locations B_1 and B_2.



(a) between $t_{\text{in}} = 0\ s$ and $t_{\text{end}} = 0.5\ s$



(b) between $t_1^{\text{zoom}} = 0.115\ s$ and $t_2^{\text{zoom}} = 0.125\ s$

Figure 2.4: Vertical accelerations at the locations B_1 and B_2.

2.3 Stability assessment with the energy method

The energy method introduced in Section 4.2.4 of Chapter 1 is here applied in order to study the stability of the *GC* algorithm. The term ${}^{n}E_{\Gamma}$ related to the work of the Lagrange multipliers and defined in Equation (1.66) as in [Gravouil and Combescure, 2001] is taken as stability indicator. If the related cumulated curve remains null or negative, the coupling does not inject energy in the system and remains stable. In particular, a negative value of the stability indicator shows the amount of numerical dissipation in the coupling.

Hereafter, the stability indicator will be normalized by the term related to the external forces defined as:

$$\sum_{n=1}^{N} {}^{n}E^{\text{ext}} = \sum_{n=1}^{N} \frac{1}{\Delta t_2} [{}^{n}\mathbf{V}_2]^T [{}^{n}\mathbf{F}_2^{\text{ext}}] \tag{2.8}$$

cumulated at the end of the analysis.

Considering the analysis introduced in Section 2.2, the normalized stability indicator is shown in Figure 2.5. The negativity of the stability indicator confirms the stability of the method demonstrated in [Gravouil and Combescure, 2001] and the numerical dissipation remarked in [Prakash and Hjelmstad, 2004] and [Mahjoubi et al., 2009]. The value decreases during the analysis reaching values close to 20% of the external forces term. The amount of numerical dissipation shown by the stability indicator is also confirmed by the results in Figure 2.4(a) in terms of accelerations: the higher frequencies at the beginning of the analysis are progressively smoothed out.

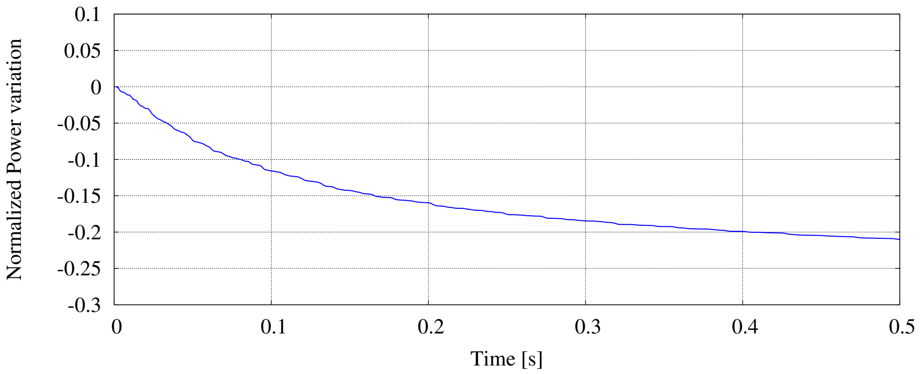
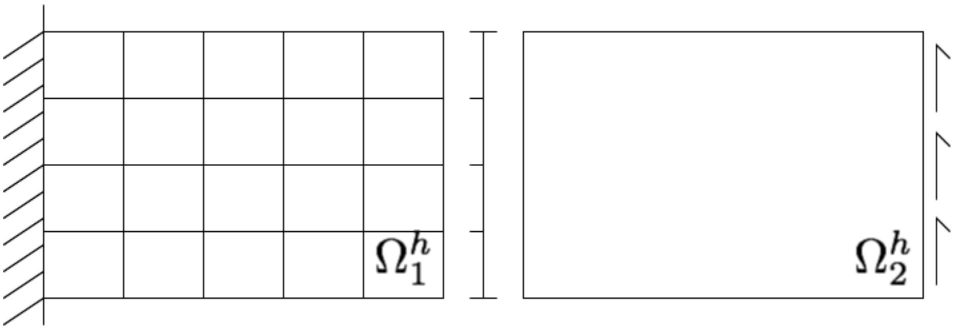


Figure 2.5: Normalized stability indicator.

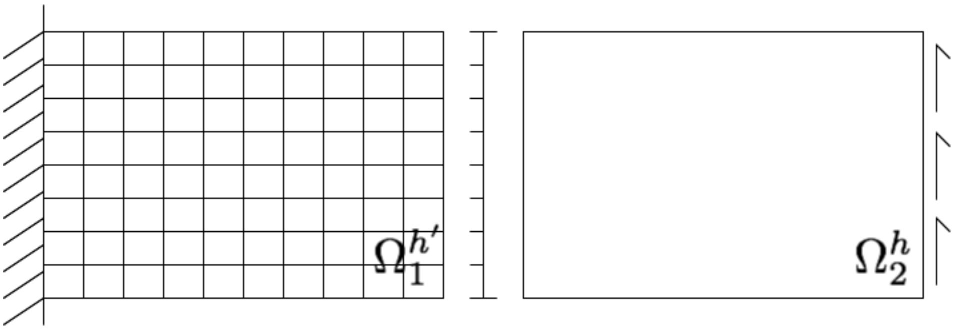
The stability indicator is then analyzed varying the discretization parameters. Figure 2.6 shows the mesh refinement applied to the first subdomain Ω_1 from Ω_1^h to $\Omega_1^{h'}$ and $\Omega_1^{h''}$ with subdomains mesh ratio $h_{2/1} = 8$ and $h_{2/1} = 16$, respectively.

Figure 2.7 shows the associated refinement of the time step Δt_1, with time steps ratio $M = 8$ and $M = 16$ for meshes $\Omega_1^{h'}$ and $\Omega_1^{h''}$, respectively.

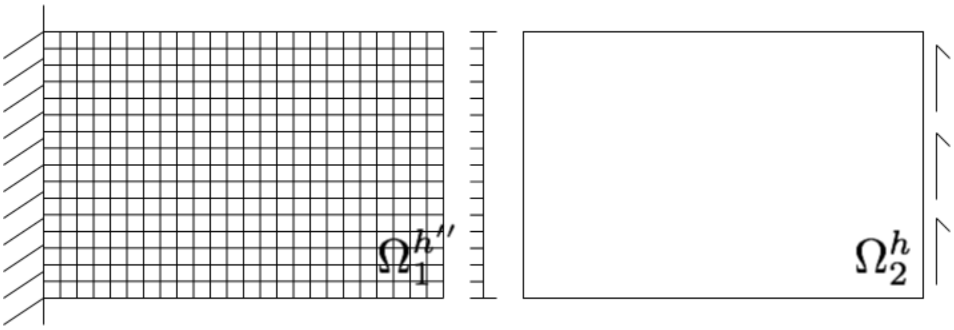
Figure 2.8 shows the variation of the stability indicator in function of the different discretization configurations, varying both the time step ratios $M = 4, 8, 16$ and the mesh ratios $h_{2/1} = 4, 8, 16$ simultaneously, keeping unchanged the mesh of the coarse subdomain



(a) subdomains mesh ratio $h_{2/1} = 4$



(b) subdomains mesh ratio $h_{2/1} = 8$



(c) subdomains mesh ratio $h_{2/1} = 16$

Figure 2.6: Refining the spatial discretization of the fine subdomain Ω_1^h.

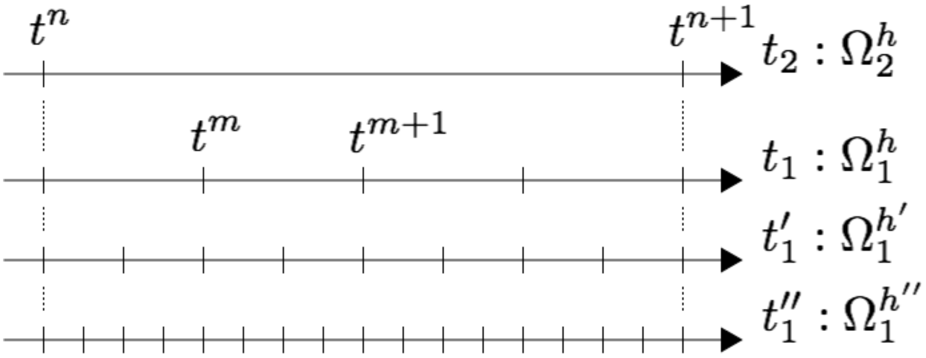


Figure 2.7: Temporal discretization related to meshes in Figure 2.6.

Ω_2^h and the related time step Δt_2. The absolute values of the stability indicator decrease with the refinement of the discretization parameters, especially passing fro $h_{2/1} = M = 4$ to $h_{2/1} = M = 8$. For this example, one can deduce that a refinement of the mesh reduces the numerical dissipation.

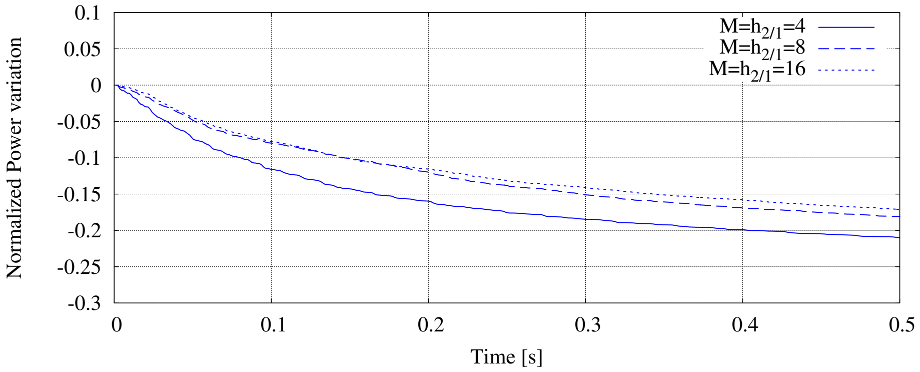


Figure 2.8: Stability indicator comparisons varying the discretization parameters $M = h_{2/1} = 4, 8, 16$ and keeping $\Delta t_2 = 12 \cdot 10^{-4}$ s.

Figure 2.9 shows the different curves of the stability indicator in function of different time step ratios $M = 4, 8, 16$, keeping unchanged the meshes Ω_1^h and Ω_2^h and the coarse time step $\Delta t_2 = 12 \cdot 10^{-4}$ s and refining Δt_1. The stability indicator curves are almost corresponding, although the bigger values of M are associated with slightly bigger indicator absolute values.

Figure 2.10 shows the stability indicator curves with the refinement of the time discretization, holding the meshes Ω_1^h and Ω_2^h and the time step ratios M. Both the time steps Δt_1 and Δt_2 are simultaneously refined. The absolute values of the stability indicator decrease with the refinement of the time steps, especially passing from $\Delta t_2 = 8 \cdot 10^{-4}$ s to $\Delta t_2 = 4 \cdot 10^{-4}$ s. It can be assessed for this example that a refinement of the time steps reduces the numerical dissipation.

Figures 2.8, 2.9 and 2.10 confirm the results reported in [Prakash and Hjelmstad,

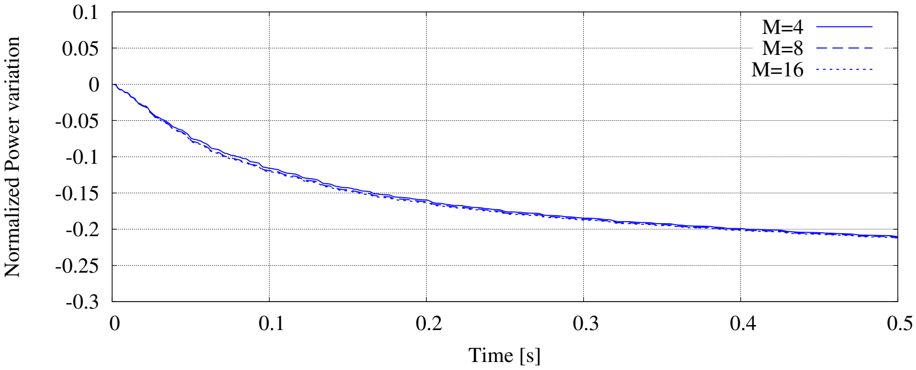


Figure 2.9: Stability indicator comparisons varying the time steps ratio $M = 4, 8, 16$, keeping $h_{2/1} = 4$ and $\Delta t_2 = 12 \cdot 10^{-4}\ s$.

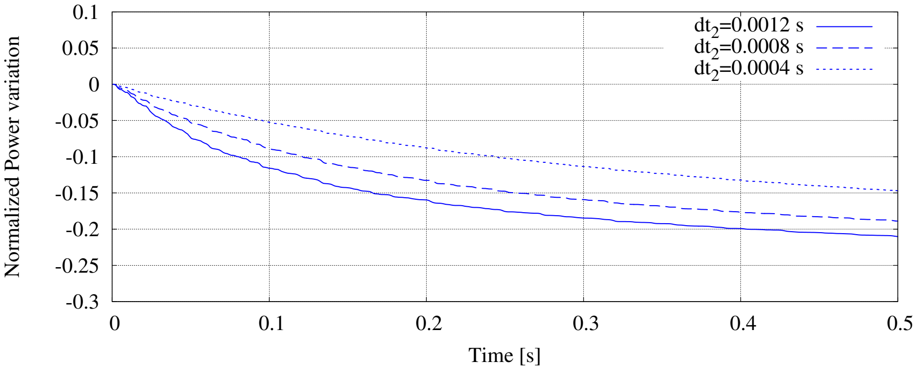


Figure 2.10: Stability indicator comparisons varying the the time steps $\Delta t_1 = 3, 1.5, 0.75 \cdot 10^{-4}\ s$ and $\Delta t_2 = 12, 8, 4 \cdot 10^{-4}\ s$, keeping $h_{2/1} = M = 4$.

2004] and [Mahjoubi et al., 2009]. Numerical dissipation due to the multi-time-step interface constraint grows without control, in contrast to the goal of the *HHT* algorithm with a numerical dissipation controlled by the parameter α (see Remark 4).

3 Energy-preserving multi-time-step FETI approach

An energy-preserving coupling method was developed in [Prakash and Hjelmstad, 2004] in order to avoid the numerical dissipation.

3.1 Formulation

System (1.57) is here redefined as introduced in Section 4.2.5 of Chapter 1, so that $\forall n \in \{0,\dots,N-1\}$:

$$\begin{array}{ll}
\mathbf{M}_1{}^{m+1}\mathbf{A}_1 + \mathbf{C}_1^T \left({}^{m+1}\mathbf{S}^{\text{free}} + {}^{n+1}\boldsymbol{\Lambda}\left(\frac{m}{M}\right)\right) = {}^{m+1}\mathbf{F}_1^{\text{ext}} - {}^{m+1}\mathbf{F}_1^{\text{int}} & \text{in } \Omega_1^h|_{t^{m+1}} \\
 & \forall m \in \{0,\dots,M-1\} \\
\mathbf{M}_2{}^{n+1}\mathbf{A}_2 - \mathbf{C}_2^T \Pi_h^{1T}\,{}^{n+1}\boldsymbol{\Lambda} = {}^{n+1}\mathbf{F}_2^{\text{ext}} - {}^{n+1}\mathbf{F}_2^{\text{int}} & \text{in } \Omega_2^h|_{t^{n+1}} \\
{}^{n+1}\mathbf{V}_{1,\Gamma} = \Pi_h^1\,{}^{n+1}\mathbf{V}_{2,\Gamma} & \text{over } \Gamma_1^h|_{t^{n+1}}
\end{array} \tag{2.9}$$

where continuity of velocities is enforced at the coarse time steps and the *free unbalanced* reaction forces ${}^{m}\mathbf{S}^{\text{free}}$ are introduced to avoid numerical dissipation.

For the sake of simplicity, the formulation will be described extending the matrix notation to the resolution of all the independent variables of acceleration, velocity and displacement during the entire coarse time step. Furthermore, the introduction of the stiffness matrix $\mathbb{K}$ is necessary for this matricial formulation, so that $\mathbf{F}^{\text{int}} = \mathbb{K}\mathbf{U}$. For a generic subdomain i:

$$\mathbb{M}_i{}^{m+1}\mathbf{X}_i = {}^{m+1}\mathbb{F}_i^{\text{ext}} - \mathbb{N}_i{}^{m}\mathbf{X}_i - \mathbb{C}_i{}^{m+1}\boldsymbol{\Lambda} \tag{2.10}$$

where:

$$\mathbb{M}_i = \begin{bmatrix} \mathbf{M}_i & \mathbf{0} & \mathbf{0} \\ -\mathbf{I}\frac{\Delta t_i}{2} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} \end{bmatrix} \quad \mathbb{N}_i = \begin{bmatrix} \mathbb{K}_i\frac{\Delta t_i^2}{2} & \mathbb{K}_i\Delta t_i & \mathbb{K}_i \\ -\mathbf{I}\frac{\Delta t_i}{2} & -\mathbf{I} & \mathbf{0} \\ -\mathbf{I}\frac{\Delta t_i^2}{2} & -\mathbf{I}\Delta t_i & -\mathbf{I} \end{bmatrix}$$

$$\mathbf{X}_i = \begin{Bmatrix} \mathbf{A}_i \\ \mathbf{V}_i \\ \mathbf{U}_i \end{Bmatrix} \quad \mathbb{F}_i^{\text{ext}} = \begin{Bmatrix} \mathbf{F}_i^{\text{ext}} \\ \mathbf{0} \\ \mathbf{0} \end{Bmatrix}$$

in which $\mathbf{I}$ is the identity matrix and, because of the application to two subdomains and embedded interface meshes:

$$\mathbb{C}_1 = \begin{bmatrix} \mathbf{C}_1^T \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} \quad \mathbb{C}_2 = \begin{bmatrix} \mathbf{C}_2^T \Pi_h^{1T} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}.$$

For each coarse time step, the resolution of the monolithic system reads:

$$
\begin{bmatrix}
\mathbb{M}_1 & & & & & \frac{1}{M}\mathbb{C}_1 \\
\mathbb{N}_1 & \mathbb{M}_1 & & & & \frac{2}{M}\mathbb{C}_1 \\
 & \ddots & \ddots & & & \vdots \\
 & & \mathbb{N}_1 & \mathbb{M}_1 & & \mathbb{C}_1 \\
 & & & & \mathbb{M}_2 & \mathbb{C}_2 \\
 & & & \mathbb{B}_1 & \mathbb{B}_2 &
\end{bmatrix}
\begin{Bmatrix}
{}^{1}\mathbf{X}_1 \\ {}^{2}\mathbf{X}_1 \\ \vdots \\ {}^{M}\mathbf{X}_1 \\ {}^{n+1}\mathbf{X}_2 \\ {}^{n+1}\boldsymbol{\Lambda}
\end{Bmatrix}
=
\begin{Bmatrix}
{}^{1}\mathbb{F}_1^{\text{ext}} - \mathbb{N}_1 {}^{n}\mathbf{X}_1 - \mathbb{C}_1 {}^{1}\mathbf{S}^{\text{free}} \\
{}^{2}\mathbb{F}_1^{\text{ext}} - \mathbb{C}_1 {}^{2}\mathbf{S}^{\text{free}} \\
\vdots \\
{}^{M}\mathbb{F}_1^{\text{ext}} - \mathbb{C}_1 {}^{M}\mathbf{S}^{\text{free}} \\
{}^{n+1}\mathbb{F}_2^{\text{ext}} - \mathbb{N}_2 {}^{n}\mathbf{X}_2 \\
\mathbf{0}
\end{Bmatrix}
\tag{2.11}
$$

where:

$$
\mathbb{B}_1 = \begin{bmatrix} \mathbf{0} & \mathbf{C}_1 & \mathbf{0} \end{bmatrix} \quad \mathbb{B}_2 = \begin{bmatrix} \mathbf{0} & \Pi_h^1 \mathbf{C}_2 & \mathbf{0} \end{bmatrix}
$$

and, as in Equation (1.68):

$$
{}^{m}\mathbf{S}^{\text{free}} = \Pi_h^1 \Pi_t^1 \left({}^{n}\mathbf{F}_{2,\Gamma}^{\text{ext}} - {}^{n}\mathbf{F}_{2,\Gamma}^{\text{int}}, {}^{n+1}\mathbf{F}_{2,\Gamma}^{\text{ext}} - {}^{n+1}\mathbf{F}_{2,\Gamma}^{\text{int}} \right) - \Pi_h^1 \mathbf{M}_{2,\Gamma} \Pi_t^1 \left({}^{n}\mathbf{A}_{2,\Gamma}, {}^{n+1}\mathbf{A}_{2,\Gamma}^{\text{free}} \right)
\tag{2.12}
$$

with the time down-scaling operator $\Pi_t^1({}^{n}\square, {}^{n+1}\square)$ as linear interpolation between ${}^{n}\square$ and ${}^{n+1}\square$.

The monolithic System (2.11) is then simplified in:

$$
\begin{bmatrix} \mathsf{M} & \mathsf{C} \\ \mathsf{B} & \mathbf{0} \end{bmatrix}
\begin{Bmatrix} {}^{n+1}\mathsf{X} \\ {}^{n+1}\boldsymbol{\Lambda} \end{Bmatrix}
=
\begin{Bmatrix} {}^{n+1}\mathsf{F} - {}^{n+1}\mathsf{S} \\ \mathbf{0} \end{Bmatrix}
\tag{2.13}
$$

where M is a lower triangle matrix with matrices $\mathbb{M}_1$, $\mathbb{M}_2$ and $\mathbb{N}_1$, C is the matrix with linear interpolation of matrices $\mathbb{C}_1$ and $\mathbb{C}_2$, B is a Boolean matrix with matrices $\mathbb{B}_1$ and $\mathbb{B}_2$, X is the vector of unknowns, ${}^{n+1}\mathsf{F}$ and ${}^{n+1}\mathsf{S}$ are the force vectors.

The *free-link* approach separates the vector of unknowns as $\mathsf{X} = \mathsf{X}^{\text{free}} + \mathsf{X}^{\text{link}}$ and System (2.13) is solved in three steps:

$$
\begin{array}{lll}
\text{1. free solution} & \mathsf{M}\,{}^{n+1}\mathsf{X}^{\text{free}} = \mathsf{F} - \mathsf{S} \\
\text{2. interface constraint} & \mathbb{H}\,{}^{n+1}\boldsymbol{\Lambda} = \mathsf{B}\,{}^{n+1}\mathsf{X}^{\text{free}} \\
\text{3. link solution} & {}^{n+1}\mathsf{X}^{\text{link}} = -\mathsf{M}^{-1}\mathsf{C}\,{}^{n+1}\boldsymbol{\Lambda}
\end{array}
\tag{2.14}
$$

where the condensation matrix $\mathbb{H}$ is defined as:

$$
\mathbb{H} \triangleq \mathsf{B}\mathbb{Y}
\tag{2.15}
$$

with the matrix $\mathbb{Y}$ calculated in order to satisfy:

$$
\mathsf{M}\mathbb{Y} = \mathsf{C}
\tag{2.16}
$$

and where:

$$
\mathsf{B}\,{}^{n+1}\mathsf{X}^{\text{free}} = \mathbf{C}_1 {}^{M}\mathbf{V}_1^{\text{free}} + \Pi_h^1 \mathbf{C}_2 {}^{n+1}\mathbf{V}_2^{\text{free}}.
\tag{2.17}
$$

Remark 7 *The condensation matrix* $\mathbb{H}$ *is a square non-diagonal matrix, whose size is equivalent to the number of nodes over the interface in the fine subdomain* Γ_1^h *multiplied by the dimension of the problem. Once again, its size is* $\dim \mathbb{H} = \mathcal{N} \dim \Gamma_1^h$. *Nevertheless, the assembly of* $\mathbb{H}$ *requires the calculation of the matrix* $\mathbb{Y}$ *via Equation* (2.16), *that is solved column by column with the resolution of the systems* $\mathsf{M}\mathbf{Y}^j = \mathbf{C}^j$, *defining the column counter* $j = 1, \dots \mathcal{N} \dim \Gamma_1^h$ *and assembling* $\mathbb{Y} = \left[\mathbf{Y}^1, \dots, \mathbf{Y}^j, \dots\right]$ *and* $\mathsf{C} = \left[\mathbf{C}^1, \dots, \mathbf{C}^j, \dots\right]$. *The matrix* M *is a lower triangle matrix, whose size is given by* $\dim \mathsf{M} = 3\mathcal{N}\left(M \dim \Omega_1^h + \dim \Omega_2^h\right)$ *where 3 accounts for displacements, velocities and accelerations. The matrix* $\mathbb{Y}$ *is also exploited in the step 3 of System* (2.14).

Algorithm 7 shows in detail the steps of the *PH* algorithm (underlined in blue the differences in comparison with the *GC* algorithm).

```
resolution of $\mathsf{M}\mathbb{Y} = \mathsf{C}$;
computation of $\mathbb{H} = \mathsf{B}\mathbb{Y}$;
while $t_2 < t_{end}$ do
    coarse free solution: $\mathbf{M}_2 {}^{n+1}\mathbf{A}_2^{\text{free}} = {}^{n+1}\mathbf{F}_2^{\text{ext}} - {}^{n+1}\mathbf{F}_2^{\text{int}}$;
    while $t_1^{free} < t_2 + \Delta t_2$ do
        compute ${}^{m+1}\mathbf{S}^{\text{free}}$ with Equation (2.12);
        fine free solution:
        $\mathbf{M}_1 {}^{m+1}\mathbf{A}_1^{\text{free}} = {}^{m+1}\mathbf{F}_1^{\text{ext}} - {}^{m+1}\mathbf{F}_1^{\text{int,free}} - \mathbf{C}_1^T \, {}^{m+1}\mathbf{S}^{\text{free}}$;
        update $t_1^{\text{free}} \leftarrow t_1^{\text{free}} + \Delta t_1$;
    end
    interface constraint: $\mathbb{H} \, {}^{n+1}\mathbf{\Lambda} = {}^{n+1}\mathbf{V}_{1,\Gamma}^{\text{free}} - \Pi_h^1 \, {}^{n+1}\mathbf{V}_{2,\Gamma}^{\text{free}}$;
    coarse link solution: $\mathbf{M}_{2,\Gamma} {}^{n+1}\mathbf{A}_{2,\Gamma}^{\text{link}} = \Pi_h^{1T} \, {}^{n+1}\mathbf{\Lambda}$;
    while $t_1^{link} < t_2 + \Delta t_2$ do
        fine link solution: $\mathbf{M}_1 {}^{m+1}\mathbf{A}_1^{\text{link}} = -\mathbf{C}_1^T \, {}^{n+1}\mathbf{\Lambda}\left(\frac{m}{M}\right) - {}^{m+1}\mathbf{F}_1^{\text{int,link}}$;
        update $t_1^{\text{link}} \leftarrow t_1^{\text{link}} + \Delta t_1$;
    end
    update $t_2 \leftarrow t_2 + \Delta t_2$;
end
```

Algorithm 7: PH algorithm.

Note that *link* solutions of the different subdomains only can be computed in parallel.

As introduced in Section 4.2.5 of Chapter 1, the *GCbis* algorithm replaces the computation of the *unbalanced free* reaction forces ${}^{m}\mathbf{S}^{\text{free}}$ with the first part of the time interpolation of the Lagrange multipliers $\left(1-\frac{m}{M}\right){}^{n}\boldsymbol{\Lambda}$, so that System (2.11) becomes:

$$\begin{bmatrix} \mathbb{M}_1 & & & & & \frac{1}{M}\mathbb{C}_1 \\ \mathbb{N}_1 & \mathbb{M}_1 & & & & \frac{2}{M}\mathbb{C}_1 \\ & \ddots & \ddots & & & \vdots \\ & & \mathbb{N}_1 & \mathbb{M}_1 & & \mathbb{C}_1 \\ & & & & \mathbb{M}_2 & \mathbb{C}_2 \\ & & & \mathbb{B}_1 & \mathbb{B}_2 & \end{bmatrix} \begin{Bmatrix} {}^{1}\mathbf{X}_1 \\ {}^{2}\mathbf{X}_1 \\ \vdots \\ {}^{M}\mathbf{X}_1 \\ {}^{n+1}\mathbf{X}_2 \\ {}^{n+1}\boldsymbol{\Lambda} \end{Bmatrix} = \begin{Bmatrix} {}^{1}\mathbb{F}_1^{\text{ext}} - \mathbb{N}_1{}^{n}\mathbf{X}_1 - \frac{M-1}{M}\mathbb{C}_1{}^{n}\boldsymbol{\Lambda} \\ {}^{2}\mathbb{F}_1^{\text{ext}} - \frac{M-2}{M}\mathbb{C}_1{}^{n}\boldsymbol{\Lambda} \\ \vdots \\ {}^{M}\mathbb{F}_1^{\text{ext}} \\ {}^{n+1}\mathbb{F}_2^{\text{ext}} - \mathbb{N}_2{}^{n}\mathbf{X}_2 \\ \mathbf{0} \end{Bmatrix}. \tag{2.18}$$

This change is more practical and efficient and maintains the energy-preserving characteristic. Algorithm 8 shows the steps of the *GCbis* algorithm (underlined in blue the differences in comparison with the *PH* algorithm).

```
resolution of $\mathsf{M}\mathbb{Y} = \mathsf{C}$;
computation of $\mathbb{H} = \mathsf{B}\mathbb{Y}$;
while $t_2 < t_{end}$ do
    coarse free solution: $\mathbf{M}_2{}^{n+1}\mathbf{A}_2^{\text{free}} = {}^{n+1}\mathbf{F}_2^{\text{ext}} - {}^{n+1}\mathbf{F}_2^{\text{int}}$;
    while $t_1^{free} < t_2 + \Delta t_2$ do
        fine free solution:
        $\mathbf{M}_1{}^{m+1}\mathbf{A}_1^{\text{free}} = {}^{m+1}\mathbf{F}_1^{\text{ext}} - {}^{m+1}\mathbf{F}_1^{\text{int,free}} - \mathbf{C}_1^{T}\,{}^{n}\boldsymbol{\Lambda}\left(1-\frac{m}{M}\right)$;
        update $t_1^{\text{free}} \leftarrow t_1^{\text{free}} + \Delta t_1$;
    end
    interface constraint: $\mathbb{H}\,{}^{n+1}\boldsymbol{\Lambda} = {}^{n+1}\mathbf{V}_{1,\Gamma}^{\text{free}} - \Pi_h^1\,{}^{n+1}\mathbf{V}_{2,\Gamma}^{\text{free}}$;
    coarse link solution: $\mathbf{M}_{2,\Gamma}{}^{n+1}\mathbf{A}_{2,\Gamma}^{\text{link}} = \Pi_h^{1T}\,{}^{n+1}\boldsymbol{\Lambda}$;
    while $t_1^{link} < t_2 + \Delta t_2$ do
        fine link solution: $\mathbf{M}_1{}^{m+1}\mathbf{A}_1^{\text{link}} = -\mathbf{C}_1^{T}\,{}^{n+1}\boldsymbol{\Lambda}\left(\frac{m}{M}\right) - {}^{m+1}\mathbf{F}_1^{\text{int,link}}$;
        update $t_1^{\text{link}} \leftarrow t_1^{\text{link}} + \Delta t_1$;
    end
    update $t_2 \leftarrow t_2 + \Delta t_2$;
end
```

Algorithm 8: GCbis algorithm.

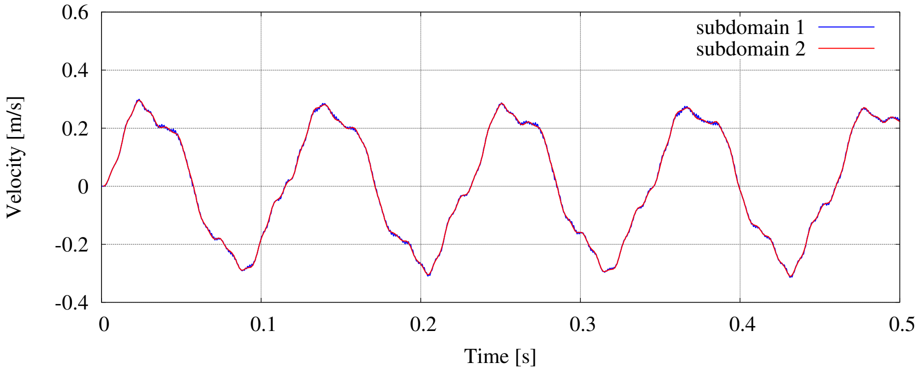
Note that both the *free* and *link* solutions of the different subdomains can be here computed in parallel.

The extensions of both the *PH* and *GCbis* algorithms to nonlinear problems and to adaptive time steps are computationally expensive, because the assembly of the condensation matrix $\mathbb{H}$ should be re-iterated through the costly scheme described in Remark 7.

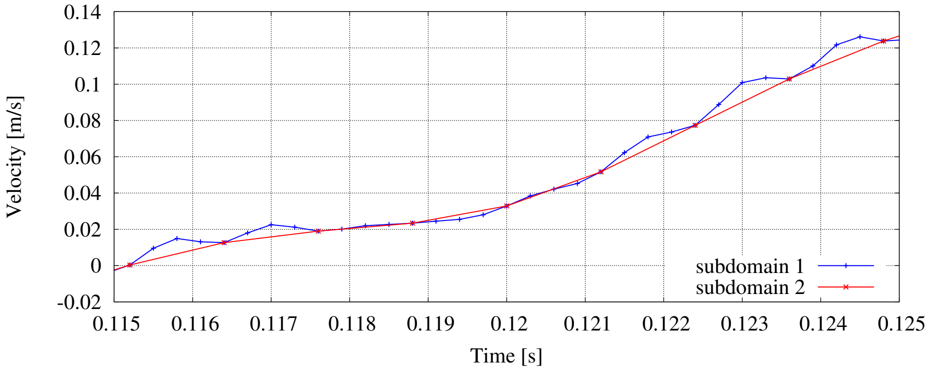
3.2 Application to assess the coupling properties

The analysis introduced in Section 2.2 is here resolved with the *GCbis* algorithm, splitting the domain into two subdomains and monitoring the four locations A, B_1, B_2 and C as in Figure 2.1.

Figure 2.11 shows the vertical velocities at the locations B_1 (subdomain 1) and B_2 (subdomain 2), considering the discretization in space and time as in Figure 2.2 with $h_{2/1} = M = 4$. According to the interface constraint, the two curves are globally overlapped in Figure 2.3(a). On the other hand, Figure 2.3(b) highlights the velocities continuity at the time instants defined in the coarse time discretization and the discrepancies in the other time instants defined with the fine time discretization, being the discrepancies due to the absence of constraint in the fine time stepping.



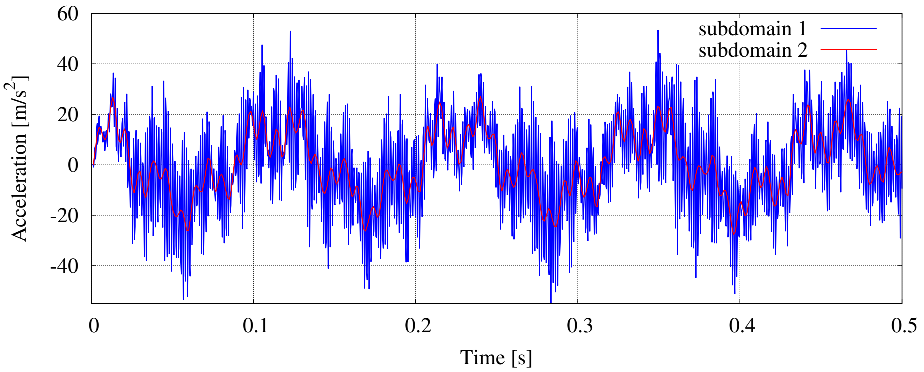
(a) between $t_{\text{in}} = 0\ s$ and $t_{\text{end}} = 0.5\ s$



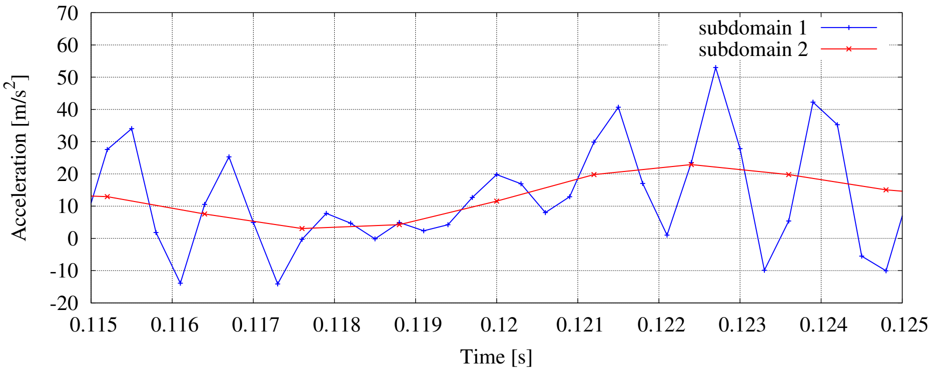
(b) between $t_1^{\text{zoom}} = 0.115\ s$ and $t_2^{\text{zoom}} = 0.125\ s$

Figure 2.11: Vertical velocities at the locations B_1 and B_2.

Figure 2.12 shows the vertical accelerations at the locations B_1 (subdomain 1) and B_2 (subdomain 2). As expected, the accelerations are not continuous. Furthermore the solution at the point B_1 in the fine subdomain oscillates differently from the analogous solution in Figure 2.4 (*GC* algorithm).



(a) between $t_{\text{in}} = 0\ s$ and $t_{\text{end}} = 0.5\ s$



(b) between $t_1^{\text{zoom}} = 0.115\ s$ and $t_2^{\text{zoom}} = 0.125\ s$

Figure 2.12: Vertical accelerations at the locations B_1 and B_2.

3.3 Conservation assessment with the energy method

The same stability indicator used in Section 2.3 is employed here in order to verify stability and conservation of the coupling algorithm, i.e.:

$$^{n}E_{\Gamma} = 0 \quad \forall n \in \{1,\dots,N\}. \tag{2.19}$$

Figure 2.13 shows the stability indicators comparing the *GC* and *GCbis* algorithms (the blue and red lines, respectively). In the comparison, the *GCbis* algorithm is denoted by a flat line of null value.

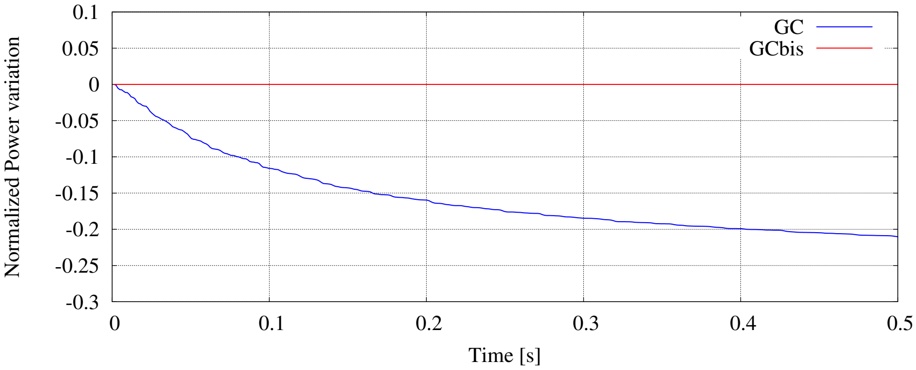


Figure 2.13: Stability indicators comparison of the GC and GCbis algorithms.

Figure 2.14 shows the values of the stability indicator, reducing the time steps and keeping the discretization ratios fixed to $h_{2/1} = M = 4$. The three curves are practically equal to zero, if one considers the machine precision used for the analysis.

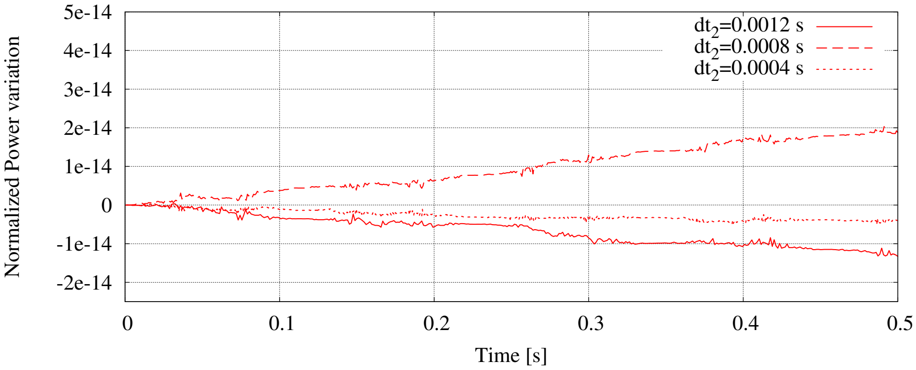


Figure 2.14: Stability indicator reducing the time steps.

4 Solution comparison between GC and GCbis algorithms

Figures 2.15 show the vertical accelerations at the location B$_2$ comparing the *GC* and *GCbis* algorithms. Figure 2.15(a) evidences the growth of the discrepancies during the analysis, due to the growing numerical dissipation of the *GC* algorithm. Figures 2.15(b-c) show the solutions of the *GC* and *GCbis* algorithms, respectively, at the beginning of the analysis, reducing the time steps keeping the discretization ratios fixed to $M = h_{2/1} = 4$. Figure 2.15(b), denoting the *GC* algorithm, shows a slower convergence reducing the time step values than Figure 2.15(c), denoting the *GCbis* algorithm.

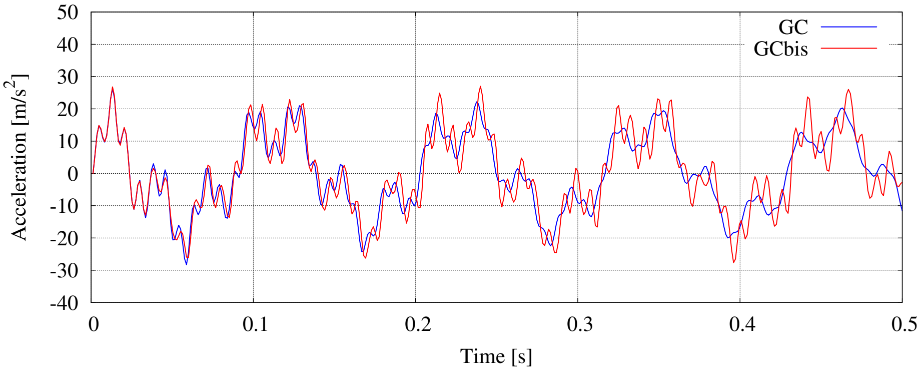
Figures 2.16 show the same comparison of the vertical accelerations at the location C at the extremity of the structure. Once again, Figure 2.15(a) evidences the growing numerical dissipation of the *GC* algorithm and Figures 2.15(b-c) show the slower convergence of the *GC* algorithm reducing the time steps, even if the *GCbis* algorithm denotes a worse solution with $\Delta t_2 = 12 \cdot 10^{-4}\ s$.

Figures 2.17 show the same comparison of the vertical accelerations at the location A inside the fine subdomain. Figure 2.15(a) evidences a high frequency solution of the *GCbis* algorithm with $\Delta t_2 = 12 \cdot 10^{-4}\ s$. Once again, Figures 2.15(b-c) show the slower convergence of the *GC* algorithm reducing the time steps. Nevertheless, Figure 2.15(c) shows that the high frequency of the solution of the *GCbis* algorithm with $\Delta t_2 = 12 \cdot 10^{-4}\ s$ is produced by a noised oscillation of the accelerations due to the too coarse time steps.

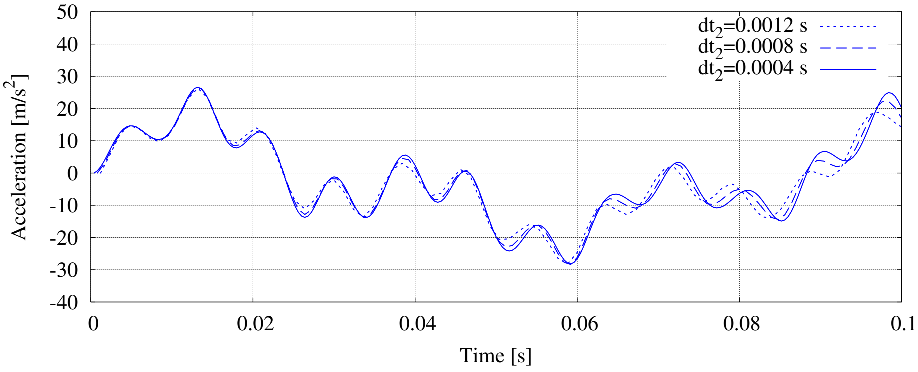
To conclude, Figures 2.15, 2.16 and 2.17 confirm the faster convergence to the exact solution of the *GCbis* algorithm reported in [Prakash and Hjelmstad, 2004] and [Mahjoubi et al., 2009]. Nevertheless, the *GCbis* algorithm expresses noisy oscillations with coarse temporal discretizations.

5 Conclusions

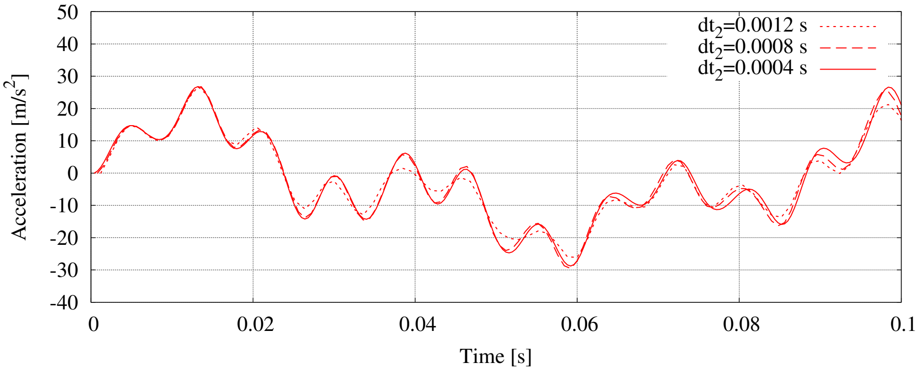
The multi-time-step FETI approach has been studied here for a linear elastic application in the evolution from the *GC* to the *GCbis* algorithms. Results show a faster convergence of the *GCbis* algorithm to the exact solution refining the time steps. Furthermore, the absence of numerical dissipation of the *GCbis* algorithm highlights its accuracy for long analyses, as highlighted in Figures 2.15(a) and 2.16(a), where the acceleration solution of the *GC* algorithm is damped out, that is why moreover referring to [Mahjoubi et al., 2009], the GCbis algorithm provides a better accuracy than the *GC* one. Therefore, the *GCbis* algorithm will be taken as reference for the development of the Substitution method.



(a) GC/GCbis algorithms comparison with $\Delta t_2 = 12 \cdot 10^{-4}\ s$

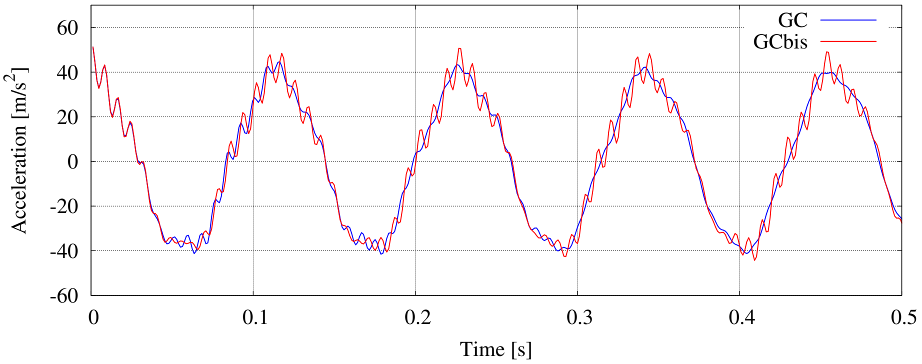


(b) GC algorithm solution at the beginning of the analysis reducing the time steps

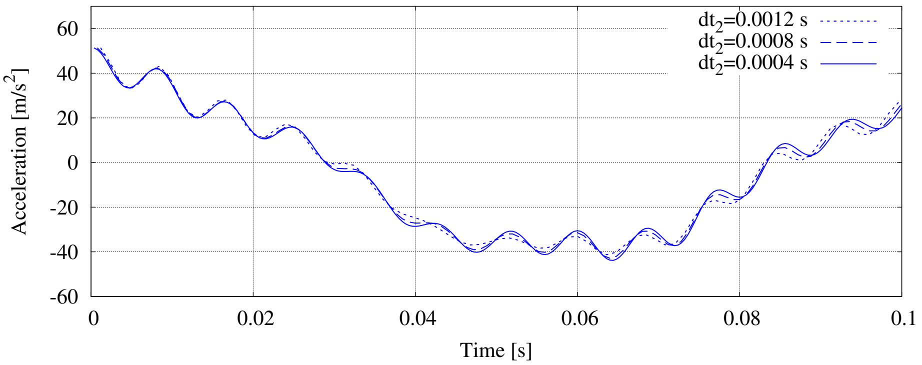


(c) GCbis algorithm solution at the beginning of the analysis reducing the time steps

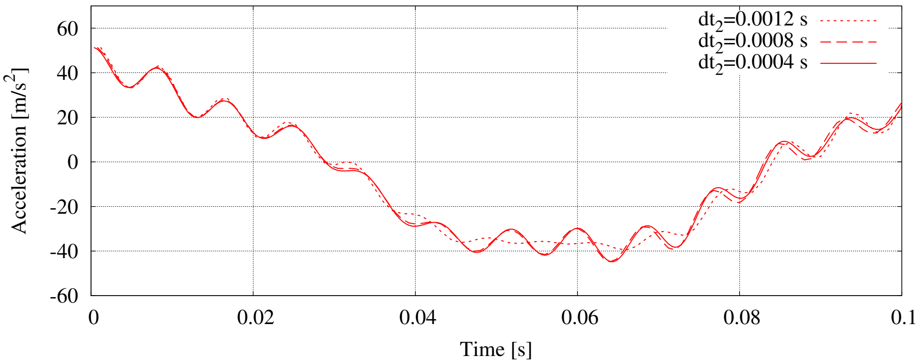
Figure 2.15: Vertical accelerations comparison of the GC and GCbis algorithms at the point B_2.



(a) GC/GCbis algorithms comparison with $\Delta t_2 = 12 \cdot 10^{-4}\ s$

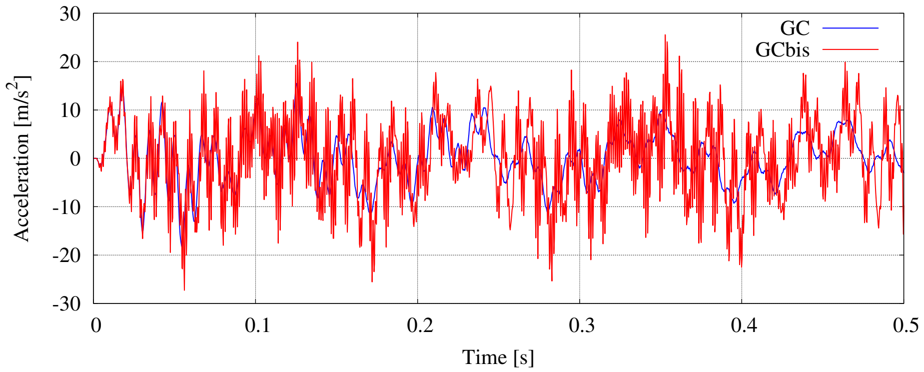


(b) GC algorithm solution at the beginning of the analysis reducing the time steps

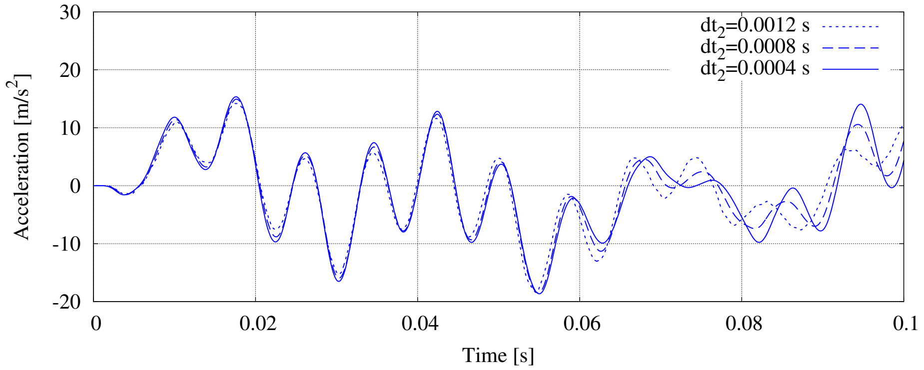


(c) GCbis algorithm solution at the beginning of the analysis reducing the time steps

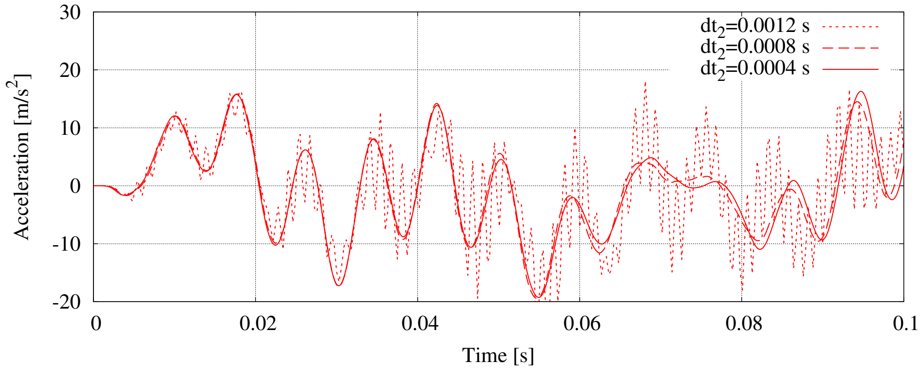
Figure 2.16: Vertical accelerations comparison of the GC and GCbis algorithms at the location C.



(a) GC/GCbis algorithms comparison with $\Delta t_2 = 12 \cdot 10^{-4}\ s$



(b) GC algorithm solution at the beginning of the analysis reducing the time steps



(c) GCbis algorithm solution at the beginning of the analysis reducing the time steps

Figure 2.17: Vertical accelerations comparison of the GC and GCbis algorithms at the location A.

Part II

Proposal and development of a weakly-intrusive multi-scale Substitution method

Chapter 3

A weakly-intrusive substitution-based coupling technique

In this Chapter as in [Bettinotti et al., 2014a], the formulation of the Substitution method is introduced and verified in comparison with the Domain Decomposition method. A stability study is performed and a substitution strategy is studied for different geometries. The implementation inside the commercial software Abaqus via the Co-Simulation technique is described and, after the extension of the Co-Simulation technique to Domain Decomposition explicit-explicit couplings, the application to truss test cases is analyzed.

Contents

1 Introduction

Non-overlapping Domain Decomposition methods constitute an efficient tool for separating the overall domain in subdomains with different models, meshes and time steps. Hereafter, the solution of such methodology, implemented as FETI multi-time-step approach with the *GC* and *GCbis* algorithms respectively in [Gravouil and Combescure, 2001] and [Mahjoubi et al., 2009], is taken as reference for the development and verification of a methodology less intrusive for dynamic adaptivity, that will be referred to as Substitution method.

Introduced in [Bettinotti et al., 2014a], the Substitution method is based on a global-local coupling formulation, with a local domain completely overlapped to the zone of interest of the global one. Local analysis patches can be activated or de-activated in function of global criteria. The global and local analyses are coupled through a kinematic constraint at the local level and an iterative correction at the global level.

The same 2-dimensional linear elastic case introduced in Chapter 1 and analyzed in Chapter 2 is here considered for the formulation of the Substitution method. A first set of analyses is then presented in comparison with the Domain Decomposition *GCbis* algorithm. The energy method is then applied to assess the stability of the method and an energy-preserving formulation is presented. As in Domain Decomposition, the extension of such methodology to deal with nonlinearities and adaptive time stepping can be computationally costly because of the required re-assembly of the condensation matrix.

New test cases are then considered. The implementation inside the commercial software Abaqus is then briefly described, with application to a simple truss test example.

2 Reference problem

To present the method, the *reference problem* introduced in Section 4 of Chapter 1 is considered. Considering Figure 1.3, the part on the left will be referred to in what follows as the *local* region and denoted by Ω_ℓ, the part on the right will be referred to as the *global complementary* region and denoted by Ω_{gc}, so that $\Omega_\ell \cup \Omega_{gc} = \Omega$.

As in Figure 1.4 in Section 4 of Chapter 1, we assume that the problem has been discretized using a non-uniform mesh of displacement-based quadrilateral elements. The partitions of the mesh are here redefined as in Figure 3.1: the fine mesh will be referred to as *local* mesh Ω_ℓ^h and the coarse mesh Ω_{gc}^h will be referred to as *global complementary* mesh.

2.1 Interface compatibility and momentum balance

The kinematic compatibility over the interface Γ^h between the two regions is enforced at the nodes of the *global complementary* mesh Ω_{gc}^h. Velocities of the remaining nodes of the interface Γ^h in the *local* mesh Ω_ℓ^h are obtained by means of linear interpolation in

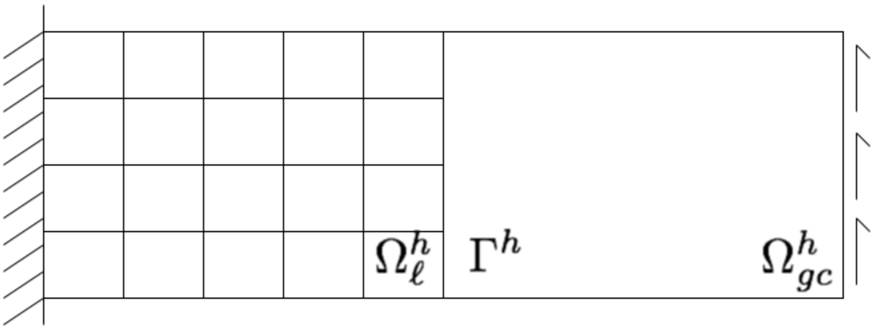


Figure 3.1: Non-uniform mesh of the reference problem.

space of velocities of the nodes of the *global complementary* mesh Ω_{gc}^h, so that:

$$\dot{\mathbf{U}}_{\ell,\Gamma} = \Pi_h^1 \dot{\mathbf{U}}_{g,\Gamma} \tag{3.1}$$

where Π_h^1 is the spatial down-scaling operator.

In addition to kinematic compatibility, the momentum balance of the interface Γ^h should be enforced as well. In order to do so, the interface Γ^h is considered in what follows as a massless entity subjected to the actions transmitted by the *local* and *global complementary* meshes Ω_ℓ^h and Ω_{gc}^h, respectively, and possibly to point or line loads directly applied to the interface Γ^h, as shown in Figure 3.2.

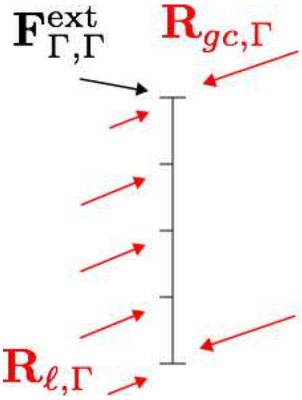


Figure 3.2: Free body diagram of interface between local and global complementary regions in the reference problem.

The momentum balance of nodes of the *global complementary* mesh Ω_{gc}^h over the interface Γ^h is so written in terms of nodal forces as follows:

$$\Pi_h^{1T} \mathbf{R}_{\ell,\Gamma} + \mathbf{R}_{gc,\Gamma} + \mathbf{F}_{\Gamma,\Gamma}^{\text{ext}} = \mathbf{0} \tag{3.2}$$

where Π_h^{1T} is the linear spatial up-scaling operator, $\mathbf{R}_{\ell,\Gamma}$ and $\mathbf{R}_{gc,\Gamma}$ are the reaction forces transmitted to the interface Γ^h by the *local* and the *global complementary* regions, respectively, as in Figure 3.3, $\mathbf{F}_{\Gamma,\Gamma}^{\text{ext}}$ are nodal forces due to possible point or line loads directly applied on the interface.

From equilibrium of the *local* and *global complementary* regions, one obtains, respectively:

$$\begin{aligned}\mathbf{R}_{\ell,\Gamma} &= \mathbf{F}^{\text{ext}}_{\ell,\Gamma} - \mathbf{M}_{\ell,\Gamma}\ddot{\mathbf{U}}_{\ell,\Gamma} - \mathbf{F}^{\text{int}}_{\ell,\Gamma} \\ \mathbf{R}_{gc,\Gamma} &= \mathbf{F}^{\text{ext}}_{gc,\Gamma} - \mathbf{M}_{gc,\Gamma}\ddot{\mathbf{U}}_{g,\Gamma} - \mathbf{F}^{\text{int}}_{gc,\Gamma}\end{aligned} \tag{3.3}$$

where $\mathbf{M}_{\ell,\Gamma}$ and $\mathbf{M}_{gc,\Gamma}$ are the lumped mass matrices, $\mathbf{F}^{\text{int}}_{\ell,\Gamma}$ and $\mathbf{F}^{\text{int}}_{gc,\Gamma}$ are internal force vectors and $\mathbf{F}^{\text{ext}}_{\ell,\Gamma}$ and $\mathbf{F}^{\text{ext}}_{gc,\Gamma}$ are the external force vectors accounting for body forces, in the *local* and *global complementary* regions, respectively.

Summarizing, at the level of the interface Γ^h, the solution of the *reference problem* is characterized by:

1. the compatibility of velocities in Equation (3.1), over the interface between *local* and *global complementary* regions;
2. the interface momentum balance in Equation (3.2), where the reaction forces over the interface, which include inertia terms, are defined by the equilibrium in Equation (3.3) of the *local* and *global complementary* regions.

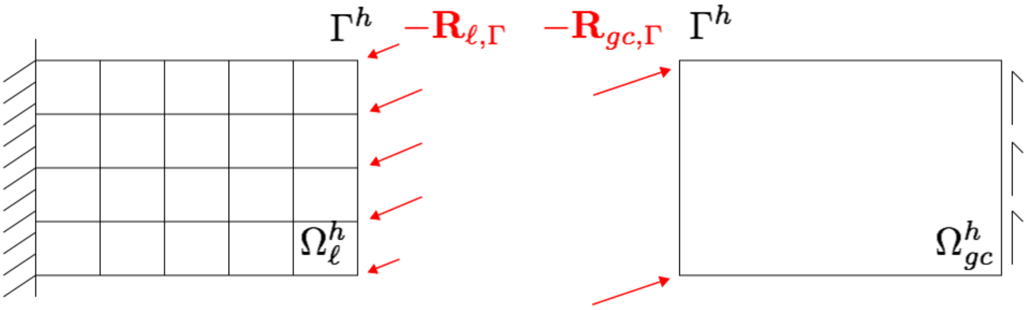


Figure 3.3: Reaction forces in local and global complementary regions.

3 Substitution method

The substitution-based coupling technique, referred to as Substitution method in the followings, aims at keeping a global analysis applied to the overall domain Ω with fixed topology during the entire analysis and with a coarse mesh Ω^h_g, correcting the coarse solution through a local analysis run on a refined model Ω^h_ℓ only where needed.

The *reference problem* in Figure 3.1 is solved by the Substitution method utilizing:

- a global analysis carried out with the coarse mesh Ω^h_g in Figure 3.4 constituted of 2 quadrilateral elements covering the entire structure;
- a local analysis carried out with the fine mesh Ω^h_ℓ in Figure 3.5 constituted of 20 quadrilateral elements and applied near the boundary conditions in order to attenuate shear locking effects.

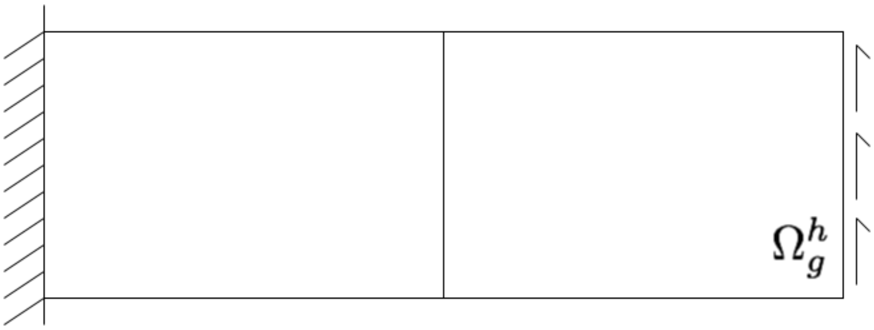


Figure 3.4: Mesh of the whole domain for the global analysis.

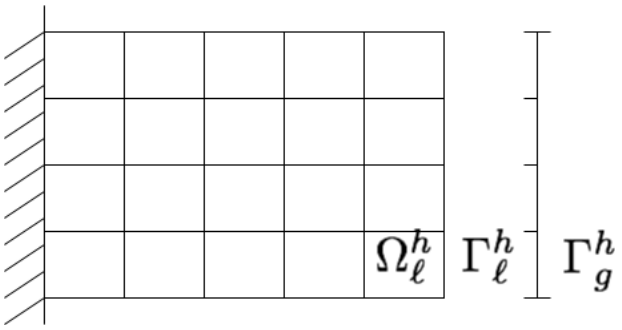


Figure 3.5: Mesh of the local region for the refined local analysis.

As described in what follows, conditions analogous to the *reference problem* in Section 2.1 ones are assumed in order to obtain same results in a semi-discrete formulation.

3.1 Interface multi-scale compatibility condition

The global-local compatibility of the velocities as in Equation (3.1) is ensured at the local level. An iterative scheme is then applied in order to reach equilibrium between the two analyses.

To formulate such conditions, the global mesh in Figure 3.4 is partitioned in *global substitution* and *global complementary* meshes as defined in Figure 3.6 with Ω_{gs}^h and Ω_{gc}^h, respectively. The *global substitution* mesh Ω_{gs}^h, which does not contain fine geometric details and can be characterized by a simplified material behavior, should be substituted by the refined *local* mesh Ω_ℓ^h in Figure 3.5.

Additional nodal forces denoted by **P**, defined in what follows as *correction forces*, have to be applied over the global interface Γ_g^h as described in Figure 3.6 in order to guarantee that the solution of the global problem is identical over the interface to the one of the *reference problem*. Such *correction forces* are intended to compensate for the different action exerted by the *local* mesh Ω_ℓ^h and by its *global substitution* Ω_{gs}^h.

The main problem in the Substitution method is then the determination of the *correction forces* **P**. Knowing those forces, the correct interface velocities can be determined

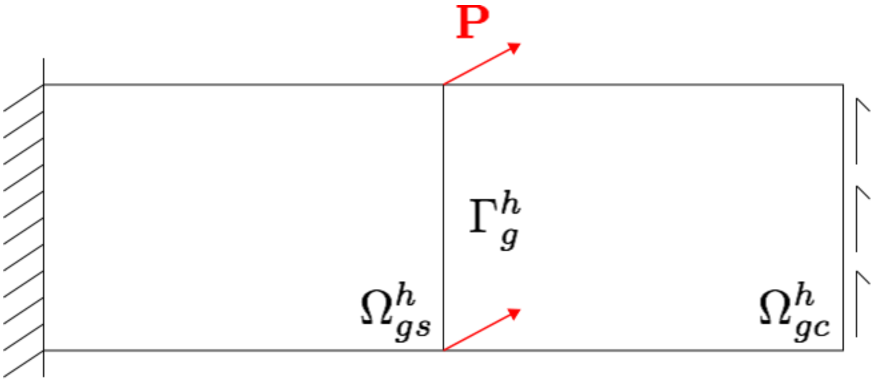


Figure 3.6: Definition of the global partition and of the correction forces **P**.

and applied to the local analysis. If the imposed velocities over the interface Γ_g^h are the correct ones, its response would then coincide with the one of the *reference problem*.

When the *local* region is replaced by the *global substitution* region and the *correction forces* are added over Γ_g^h, the *global interface momentum balance* reads:

$$\mathbf{R}_{gs,\Gamma} + \mathbf{R}_{gc,\Gamma} + \mathbf{P} + \mathbf{F}^{\text{ext}}_{\Gamma,\Gamma} = \mathbf{0} \tag{3.4}$$

where:

$$\mathbf{R}_{gs,\Gamma} = \mathbf{F}^{\text{ext}}_{gs,\Gamma} - \mathbf{M}_{gs,\Gamma}\ddot{\mathbf{U}}_{g,\Gamma} - \mathbf{F}^{\text{int}}_{gs,\Gamma}. \tag{3.5}$$

The *correction forces* **P** are determined by imposing that the left-hand sides of Equations (3.2) and (3.4) are equivalent, so that:

$$\mathbf{R}_{gs,\Gamma} + \mathbf{R}_{gc,\Gamma} + \mathbf{P} + \mathbf{F}^{\text{ext}}_{\Gamma,\Gamma} = \Pi_h^{1T}\mathbf{R}_{\ell,\Gamma} + \mathbf{R}_{gc,\Gamma} + \mathbf{F}^{\text{ext}}_{\Gamma,\Gamma}. \tag{3.6}$$

Eliminating the values $\mathbf{R}_{gc,\Gamma}$ and $\mathbf{F}^{\text{ext}}_{\Gamma,\Gamma}$, Equation (3.6) becomes:

$$\mathbf{P} = \Pi_h^{1T}\mathbf{R}_{\ell,\Gamma} - \mathbf{R}_{gs,\Gamma}, \tag{3.7}$$

i.e., using the definition of reaction forces given in Equations $(3.3)_1$ and (3.5):

$$\mathbf{P} = \Pi_h^{1T}\left(\mathbf{F}^{\text{ext}}_{\ell,\Gamma} - \mathbf{M}_{\ell,\Gamma}\ddot{\mathbf{U}}_{\ell,\Gamma} - \mathbf{F}^{\text{int}}_{\ell,\Gamma}\right) - \mathbf{F}^{\text{ext}}_{gs,\Gamma} + \mathbf{M}_{gs,\Gamma}\ddot{\mathbf{U}}_{g,\Gamma} + \mathbf{F}^{\text{int}}_{gs,\Gamma}. \tag{3.8}$$

Remark 8 *Considering the correction and reaction definitions above and knowing that over the interface:*

$$\mathbf{M}_{g,\Gamma} = \mathbf{M}_{gs,\Gamma} + \mathbf{M}_{gc,\Gamma}, \quad \mathbf{F}^{int}_{g,\Gamma} = \mathbf{F}^{int}_{gs,\Gamma} + \mathbf{F}^{int}_{gc,\Gamma}, \quad \mathbf{F}^{ext}_{g,\Gamma} = \mathbf{F}^{ext}_{gs,\Gamma} + \mathbf{F}^{ext}_{gc,\Gamma} + \mathbf{F}^{ext}_{\Gamma,\Gamma}, \tag{3.9}$$

the global interface momentum balance in Equation (3.4) *becomes:*

$$\mathbf{M}_{g,\Gamma}\ddot{\mathbf{U}}_{g,\Gamma} + \mathbf{F}^{int}_{g,\Gamma} = \mathbf{F}^{ext}_{g,\Gamma} + \mathbf{P}. \tag{3.10}$$

3.2 Global time integration with the correction forces

Discretizing the analysis time interval $[t_{\text{in}}, t_{\text{end}}]$ with an integer number N of pre-fixed global time steps Δt_g, such that $N\Delta t_g = t_{\text{end}} - t_{\text{in}}$, the integration in time of Equation (3.10) following the Central Difference scheme allows one to write the *global interface momentum balance* in the global time stepping as:

$$\mathbf{M}_{g,\Gamma}{}^{n+1}\mathbf{A}_{g,\Gamma} = {}^{n+1}\mathbf{F}^{\text{ext}}_{g,\Gamma} - {}^{n+1}\mathbf{F}^{\text{int}}_{g,\Gamma} + {}^{n+1}\mathbf{P} \tag{3.11}$$

where the *correction forces* ${}^{n+1}\mathbf{P}$ are:

$${}^{n+1}\mathbf{P} = \Pi_h^{1T}\,{}^{n+1}\mathbf{R}_{\ell,\Gamma} - {}^{n+1}\mathbf{R}_{gs,\Gamma} \tag{3.12}$$

with the global substitution and local nodal forces ${}^{n+1}\mathbf{R}_{gs,\Gamma}$ and ${}^{n+1}\mathbf{R}_{\ell,\Gamma}$ respectively defined as:

$$\begin{aligned} {}^{n+1}\mathbf{R}_{\ell,\Gamma} &= {}^{n+1}\mathbf{F}^{\text{ext}}_{\ell,\Gamma} - \mathbf{M}_{\ell,\Gamma}{}^{n+1}\mathbf{A}_{\ell,\Gamma} - {}^{n+1}\mathbf{F}^{\text{int}}_{\ell,\Gamma}, \\ {}^{n+1}\mathbf{R}_{gs,\Gamma} &= {}^{n+1}\mathbf{F}^{\text{ext}}_{gs,\Gamma} - \mathbf{M}_{gs,\Gamma}{}^{n+1}\mathbf{A}_{g,\Gamma} - {}^{n+1}\mathbf{F}^{\text{int}}_{gs,\Gamma}. \end{aligned} \tag{3.13}$$

From Equations (3.11), (3.12) and (3.13), it is highlighted that the global solution is function of the local one and vice versa, because the *correction forces* $\mathbf{P}$ are computed in function of the local solution and the global velocities are imposed over the local interface Γ_ℓ^h in the local analysis, so Equation (3.11) is a nonlinear set of algebraic equations that requires the iterative correction of the global solution in function of the local one over the local time stepping.

3.3 Iterative scheme: fixed-point algorithm

An iterative fixed-point algorithm is employed to solve Equation (3.11). At the n^{th} global time step and i^{th} iteration, the *global interface momentum balance* reads:

$$\mathbf{M}_{g,\Gamma}\left({}^{n+1}\mathbf{A}_{g,\Gamma}\right)^i = {}^{n+1}\mathbf{F}^{\text{ext}}_{g,\Gamma} - {}^{n+1}\mathbf{F}^{\text{int}}_{g,\Gamma} + \left({}^{n+1}\mathbf{P}\right)^i \tag{3.14}$$

where the *correction forces* are defined as:

$$\left({}^{n+1}\mathbf{P}\right)^i = \Pi_h^{1T}\left({}^{n+1}\mathbf{R}_{\ell,\Gamma}\right)^i - \left({}^{n+1}\mathbf{R}_{gs,\Gamma}\right)^i \tag{3.15}$$

with:

$$\begin{aligned} \left({}^{n+1}\mathbf{R}_{\ell,\Gamma}\right)^i &= {}^{n+1}\mathbf{F}^{\text{ext}}_{\ell,\Gamma} - \mathbf{M}_{\ell,\Gamma}\left({}^{n+1}\mathbf{A}_{\ell,\Gamma}\right)^i - \left({}^{n+1}\mathbf{F}^{\text{int}}_{\ell,\Gamma}\right)^i \\ \left({}^{n+1}\mathbf{R}_{gs,\Gamma}\right)^i &= {}^{n+1}\mathbf{F}^{\text{ext}}_{gs,\Gamma} - \mathbf{M}_{gs,\Gamma}\left({}^{n+1}\mathbf{A}_{g,\Gamma}\right)^{i-1} - {}^{n+1}\mathbf{F}^{\text{int}}_{gs,\Gamma} \end{aligned} \tag{3.16}$$

in which the global substitution reaction forces are expressed in function of the previous iteration $i-1$, that must be known to be placed in the right-hand side of Equation (3.14).

Thus, Equation (3.14) can be rewritten by developing the corrective term $\left({}^{n+1}\mathbf{P}\right)^i$ as:

$$\mathbf{M}_{g,\Gamma}\left({}^{n+1}\mathbf{A}_{g,\Gamma}\right)^i = {}^{n+1}\mathbf{F}^{\text{ext}}_{g,\Gamma} - {}^{n+1}\mathbf{F}^{\text{int}}_{g,\Gamma} + \Pi_h^{1T}\left({}^{n+1}\mathbf{R}_{\ell,\Gamma}\right)^i - \left({}^{n+1}\mathbf{R}_{gs,\Gamma}\right)^i \tag{3.17}$$

that, developing the global reaction forces $\left({}^{n+1}\mathbf{R}_{gs,\Gamma}\right)^i$ defined in Equation $(3.13)_2$ and considering the splitting definitions in Equations (3.9), becomes:

$$\mathbf{M}_{g,\Gamma}\left({}^{n+1}\mathbf{A}_{g,\Gamma}\right)^i = {}^{n+1}\mathbf{F}^{\text{ext}}_{\Gamma,\Gamma} + {}^{n+1}\mathbf{F}^{\text{ext}}_{gc,\Gamma} - {}^{n+1}\mathbf{F}^{\text{int}}_{gc,\Gamma} + \Pi_h^{1T}\left({}^{n+1}\mathbf{R}_{\ell,\Gamma}\right)^i + \mathbf{M}_{gs,\Gamma}\left({}^{n+1}\mathbf{A}_{g,\Gamma}\right)^{i-1}. \tag{3.18}$$

Defining the *global complementary* reaction forces $\left({}^{n+1}\mathbf{R}_{gc}\right)^i$ again in function of the previous iteration $i-1$:

$$\left({}^{n+1}\mathbf{R}_{gc}\right)^i = {}^{n+1}\mathbf{F}^{\text{ext}}_{gc,\Gamma} - \mathbf{M}_{gc,\Gamma}\left({}^{n+1}\mathbf{A}_{g,\Gamma}\right)^{i-1} - {}^{n+1}\mathbf{F}^{\text{int}}_{gc,\Gamma} \tag{3.19}$$

Equation (3.18) becomes:

$$\mathbf{M}_{g,\Gamma}\left({}^{n+1}\mathbf{A}_{g,\Gamma}\right)^i = \Pi_h^{1T}\left({}^{n+1}\mathbf{R}_{\ell,\Gamma}\right)^i + \left({}^{n+1}\mathbf{R}_{gc,\Gamma}\right)^i + {}^{n+1}\mathbf{F}^{\text{ext}}_{\Gamma,\Gamma} + \mathbf{M}_{g,\Gamma}\left({}^{n+1}\mathbf{A}_{g,\Gamma}\right)^{i-1} \tag{3.20}$$

that implies for $i \to +\infty$:

$$\left({}^{n+1}\mathbf{A}_{g,\Gamma}\right)^i \to \left({}^{n+1}\mathbf{A}_{g,\Gamma}\right)^{i-1} \quad \Leftrightarrow \quad \Pi_h^{1T}\left({}^{n+1}\mathbf{R}_{\ell,\Gamma}\right)^i + \left({}^{n+1}\mathbf{R}_{gc,\Gamma}\right)^i + {}^{n+1}\mathbf{F}^{\text{ext}}_{\Gamma,\Gamma} \to \mathbf{0} \tag{3.21}$$

i.e. stagnation of the solution in terms of accelerations implies equilibrium over the interface.

The measure of the convergence of the iterative scheme is then taken as:

$$e = \frac{\left\|\Pi_h^{1T}\left({}^{n+1}\mathbf{R}_{\ell}\right)^i + \left({}^{n+1}\mathbf{R}_{gc}\right)^i + {}^{n+1}\mathbf{F}^{\text{ext}}_{\Gamma,\Gamma}\right\|_{L^2(\Gamma_g^h)}}{\left\|{}^{N}\mathbf{F}^{\text{ext}}_{g}\right\|_{L^2(\partial\Omega_f)}} \tag{3.22}$$

where $\|\square\|_{L^2(\square)}$ represents an L^2 norm over the surface $\square$ and N denotes the final time counter. For more general purposes, an other type of normalization of the convergence measure should be considered.

3.4 Time down-scaling operator

Nonlinearities in Equation (3.11) are so due to the dependencies between the global and local analyses. Indeed, the local analysis is run applying the continuity between global and local velocities over the interface, as compatibility condition of the *reference problem* in Equation (3.1). Nevertheless, the global and local time scales are different and a time down-scaling operator is necessary from global to local velocities. Two different and pre-fixed time steps are defined in the global and local analyses such that $\Delta t_g = M\Delta t_\ell$, as illustrated in Figure 3.7.

The integration of Equation (3.1) in time forces one to define a down-scaling time operator $\Pi_t(\square)$ in order to impose the kinematic constraint at the local time steps between t^n and t^{n+1}, so that:

$$\dot{\mathbf{U}}_{\ell,\Gamma}\big|_{[t_{\text{in}},t_{\text{end}}]} = \Pi_h^1\Pi_t\left(\dot{\mathbf{U}}_{g,\Gamma}\right). \tag{3.23}$$

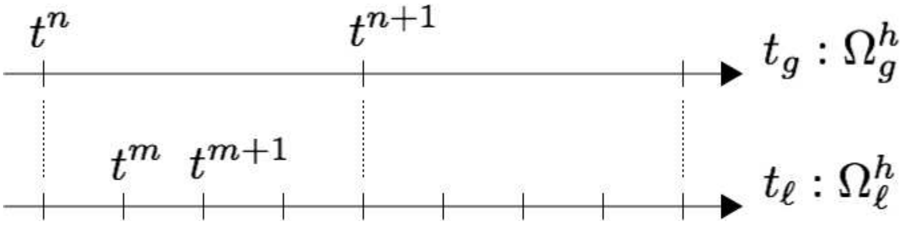


Figure 3.7: Time discretization for global and local analyses.

3.4.1 Linear interpolation

As time down-scaling operator of Domain Decomposition *GC* algorithm, the linear interpolation $\Pi_t^1({}^n\square, {}^{n+1}\square)$ of velocities is tested for the Substitution method. The local constraint reads:

$$\dot{\mathbf{U}}_{\ell,\Gamma}(s)|_{[t^n,t^{n+1}]} = {}^n\mathbf{V}_{g,\Gamma}(1-s) + {}^{n+1}\mathbf{V}_{g,\Gamma}\, s \tag{3.24}$$

with $s(t) = (t - t^n)/\Delta t_g$ and $s \in [0,1]$. Considering the local time-stepping with $s(t^m)$:

$${}^m\mathbf{V}_{\ell,\Gamma} = {}^n\mathbf{V}_{g,\Gamma}\left(1 - \frac{m}{M}\right) + {}^{n+1}\mathbf{V}_{g,\Gamma}\left(\frac{m}{M}\right). \tag{3.25}$$

In the simple example proposed in Figure 3.8, the motion of a point with mass $\mathcal{M}$ constrained by a spring of stiffness $\mathcal{K}$ and perturbed by a force constant in time is described with the Central Difference scheme as global analysis, whose solution will be denoted by the subscript "$g,\circ$". The same motion in terms of velocities is down-scaled to a finer time scale, such that the time step ratio $M = 10$, and the Central Difference scheme is used to calculate displacements and accelerations as local analysis: these variables will be denoted by the subscript "$\ell,\circ$".

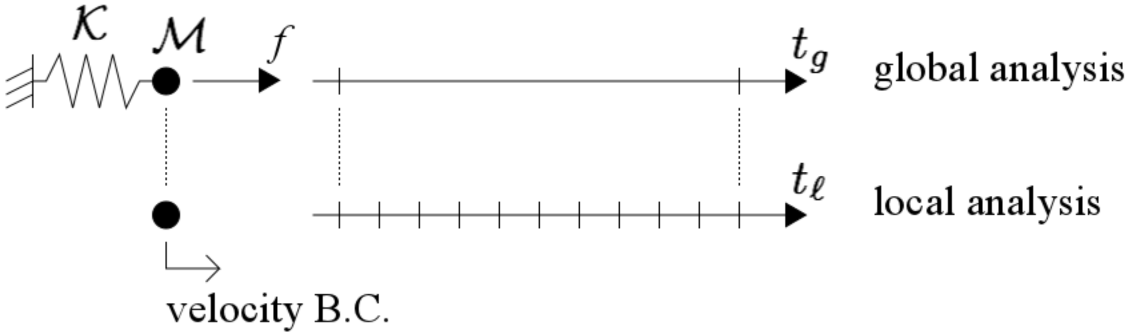


Figure 3.8: Spring example for testing the time down-scaling operator, taking a time step ratio $M = 10$.

As shown in Figure 3.9, velocities are linearly interpolated and are continuous at each local time step, whereas accelerations jump between two values in the global time step.

With the linear interpolation in Equation (3.25), one would thus have at each local time step:

$${}^{m+1}\mathbf{V}_{\ell,\circ} - {}^m\mathbf{V}_{\ell,\circ} = \left({}^{n+1}\mathbf{V}_{g,\circ} - {}^n\mathbf{V}_{g,\circ}\right)/M \tag{3.26}$$

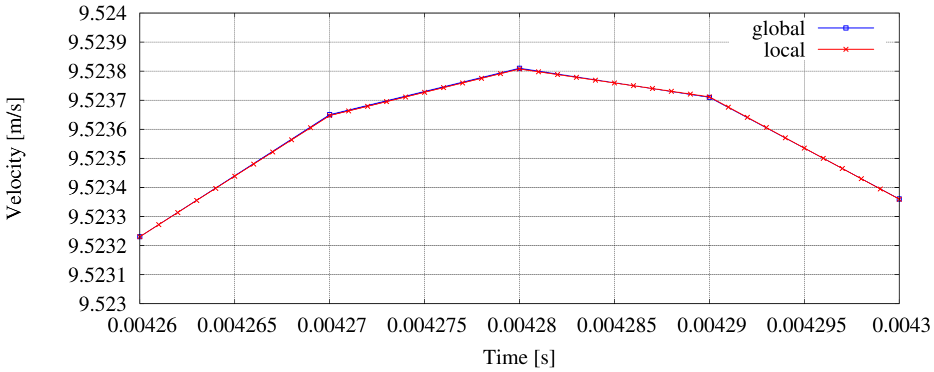


Figure 3.9: Velocity constraint as a linear interpolation in time.

which is constant throughout the local analysis in the global time step.

From the Central Difference scheme applied to both the global and local analyses:

$$\begin{aligned} {}^{m+1}\mathbf{V}_{\ell,\circ} - {}^{m}\mathbf{V}_{\ell,\circ} &= \left({}^{m+1}\mathbf{A}_{\ell,\circ} + {}^{m}\mathbf{A}_{\ell,\circ}\right)\frac{\Delta t_\ell}{2} \\ {}^{n+1}\mathbf{V}_{g,\circ} - {}^{n}\mathbf{V}_{g,\circ} &= \left({}^{n+1}\mathbf{A}_{g,\circ} + {}^{n}\mathbf{A}_{g,\circ}\right)\frac{\Delta t_g}{2} \end{aligned} \tag{3.27}$$

and Equation (3.26) becomes:

$$\left({}^{m+1}\mathbf{A}_{\ell,\circ} + {}^{m}\mathbf{A}_{\ell,\circ}\right)\frac{\Delta t_\ell}{2} = \left({}^{n+1}\mathbf{A}_{g,\circ} + {}^{n}\mathbf{A}_{g,\circ}\right)\frac{\Delta t_g}{2M} \tag{3.28}$$

i.e., with the definition of the global time step $\Delta t_g = M\Delta t_\ell$:

$${}^{m+1}\mathbf{A}_{\ell,\circ} + {}^{m}\mathbf{A}_{\ell,\circ} = {}^{n+1}\mathbf{A}_{g,\circ} + {}^{n}\mathbf{A}_{g,\circ}. \tag{3.29}$$

As a result of the linear interpolation of global velocities, Equation (3.29) is enforced at each local time step of the entire local analysis. Therefore, assuming homogeneous initial conditions ${}^{0}\mathbf{A}_{\ell,\circ} = {}^{0}\mathbf{A}_{g,\circ} = \mathbf{0}$:

- if M is odd, ${}^{m}\mathbf{A}_{\ell,\circ}$ jumps between ${}^{n}\mathbf{A}_{g,\circ}$ and ${}^{n+1}\mathbf{A}_{g,\circ}$ with $m = 1, \ldots, M$,
- if M is even, ${}^{m}\mathbf{A}_{\ell,\circ}$ jumps between $\mathbf{0}$ and $\left({}^{n}\mathbf{A}_{g,\circ} + {}^{n+1}\mathbf{A}_{g,\circ}\right)$ with $m = 1, \ldots, M$, as illustrated in Figure 3.10 with $M = 10$.

Such jumping behavior leads to unstable and inconsistent results. Therefore, an alternative to the linear interpolation of global velocities in the local time stepping has so to be considered, in order to avoid such numerical outcome.

3.4.2 Cubic Hermitian interpolation

The idea to overcome the preliminary difficulties is so to use more informations from the global solution in the local constraint.

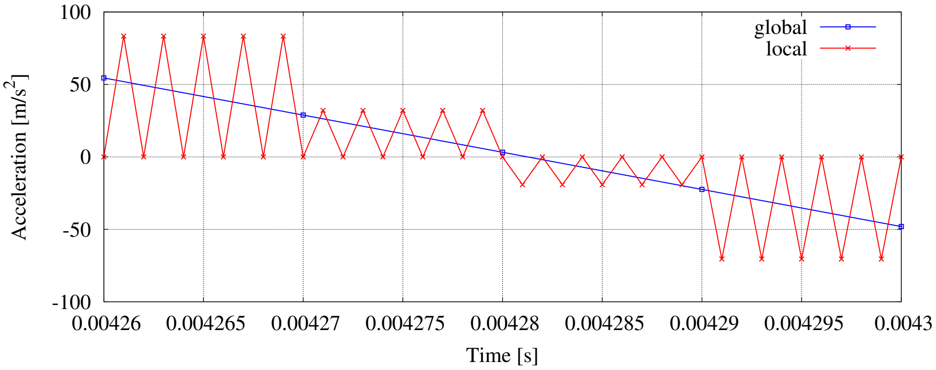


Figure 3.10: Resulting accelerations with a linear interpolation of velocity in time and applying the Central Difference scheme to global and local solutions.

In this way, as an alternative to the linear interpolation $\Pi_t^1({}^n\square, {}^{n+1}\square)$ in Equation (3.24), the cubic Hermitian interpolation $\Pi_t^3({}^n\square, {}^n\triangle, {}^{n+1}\square, {}^{n+1}\triangle)$ is defined in function of both global velocities and accelerations, so that:

$$\dot{\mathbf{U}}_{\ell,\Gamma}\big|_{[t^n,t^{n+1}]} = \Pi_h^1 \Pi_t^3 \left({}^n\mathbf{V}_{g,\Gamma}, {}^n\mathbf{A}_{g,\Gamma}, {}^{n+1}\mathbf{V}_{g,\Gamma}, {}^{n+1}\mathbf{A}_{g,\Gamma}\right) \tag{3.30}$$

that, introducing the intrinsic local time $s(t)$ and 4 cubic polynomials $\psi_i(s)$, becomes:

$$\dot{\mathbf{U}}_{\ell,\Gamma}\big|_{[t^n,t^{n+1}]} = \Pi_h^1 \left({}^n\mathbf{V}_{g,\Gamma}\psi_1(s) + {}^n\mathbf{A}_{g,\Gamma}\Delta t_g \psi_2(s) + {}^{n+1}\mathbf{V}_{g,\Gamma}\psi_3(s) + {}^{n+1}\mathbf{A}_{g,\Gamma}\Delta t_g \psi_4(s)\right). \tag{3.31}$$

The 4 cubic polynomials $\psi_i(s)$ with $i = 1, \ldots, 4$ are obtained imposing:

$$\psi_i(0) = \delta_{i1}, \quad \dot{\psi}_i(0) = \delta_{i2}, \quad \psi_i(1) = \delta_{i3}, \quad \dot{\psi}_i(1) = \delta_{i4}, \quad i = 1, \ldots 4 \tag{3.32}$$

where δ_{ij} is the Kronecker's delta, so that:

$$\begin{aligned}\dot{\mathbf{U}}_{\ell,\Gamma}\big|_{[t^n,t^{n+1}]} = {} & \Pi_h^1\, {}^n\mathbf{V}_{g,\Gamma}\left(1 - 3s^2 + 2s^3\right) + \Pi_h^1\, {}^n\mathbf{A}_{g,\Gamma}\Delta t_g\left(s - 2s^2 + s^3\right) + \\ & + \Pi_h^1\, {}^{n+1}\mathbf{V}_{g,\Gamma}\left(3s^2 - 2s^3\right) + \Pi_h^1\, {}^{n+1}\mathbf{A}_{g,\Gamma}\Delta t_g\left(-s^2 + s^3\right)\end{aligned} \tag{3.33}$$

and considering the local time-stepping:

$$\begin{aligned}{}^m\mathbf{V}_{\ell,\Gamma} = {} & \Pi_h^1\, {}^n\mathbf{V}_{g,\Gamma}\left(1 - 3\frac{m^2}{M^2} + 2\frac{m^3}{M^3}\right) + \Pi_h^1\, {}^n\mathbf{A}_{g,\Gamma}\Delta t_g\left(\frac{m^2}{M^2} - 2\frac{m^2}{M^2} + \frac{m^3}{M^3}\right) + \\ & + \Pi_h^1\, {}^{n+1}\mathbf{V}_{g,\Gamma}\left(3\frac{m^2}{M^2} - 2\frac{m^3}{M^3}\right) + \Pi_h^1\, {}^{n+1}\mathbf{A}_{g,\Gamma}\Delta t_g\left(-\frac{m^2}{M^2} + \frac{m^3}{M^3}\right)\end{aligned} \tag{3.34}$$

For the simple example of Figure 3.8, a continuity at the global time steps between the global and local velocities and between the global and local accelerations is expressed, as shown in Figures 4.1.

On the other hand, as it was found out for Domain Decomposition with multi-time-step coupling, the displacements continuity is not ensured, as highlighted in Figure 3.12.

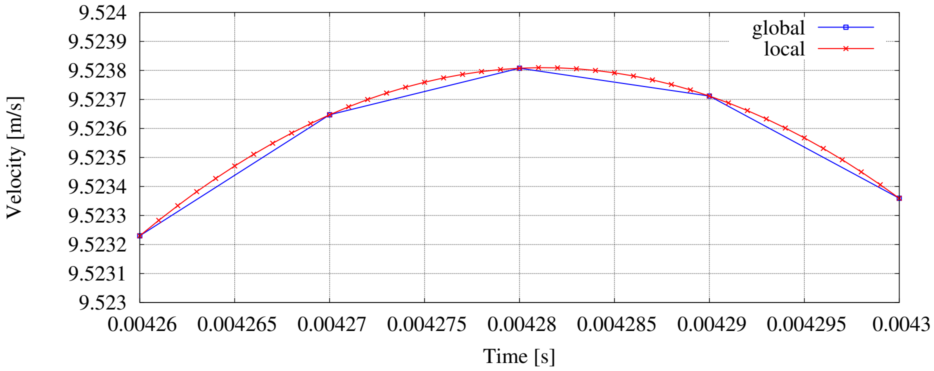


Figure 3.11: Velocity constraint as a cubic Hermitian interpolation in time in function of global velocities and accelerations.

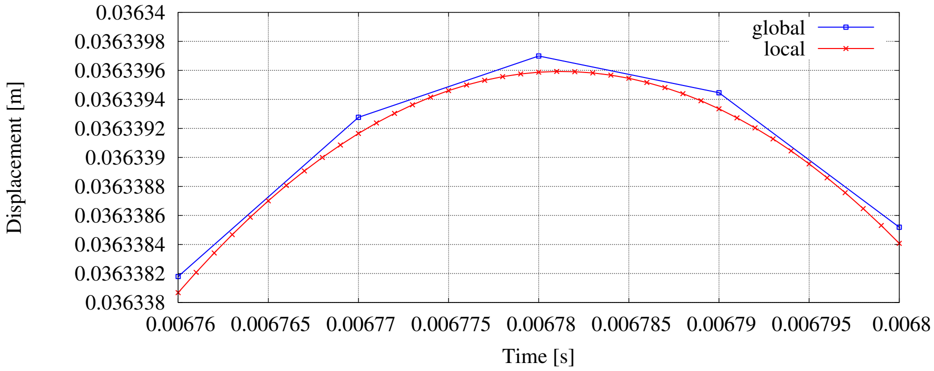


Figure 3.12: Displacements discontinuity with a cubic Hermitian interpolation in time and applying the Central Difference scheme for global and local solutions.

3.5 Substitution strategy

Combining the iterative scheme of Equations (3.14), (3.15) and (3.16) in Section 3.3 with the cubic Hermitian interpolation in Equation (3.34) as time down-scaling operator in Section 3.4, the Substitution method for the application in Figure 1.3 makes use of:

1. a global analysis in the mesh Ω_g^h defined in Figure 3.4, so that $\forall n \in \{0,\ldots,N-1\}$:

$$\mathbf{M}_g \left({}^{n+1}\mathbf{A}_g\right)^i = {}^{n+1}\mathbf{F}_g^{\text{ext}} - {}^{n+1}\mathbf{F}_g^{\text{int}} + \mathbf{C}_g^T \left({}^{n+1}\mathbf{P}\right)^i \tag{3.35}$$

   where $\mathbf{C}_g$ is the Boolean unitary matrix that extracts nodal values from the global domain to the global interface, so that:

$$\mathbf{C}_{g,(\dim\Gamma_g^h \times \dim\Omega_g^h)} : \Omega_g^h \mapsto \Gamma_g^h \tag{3.36}$$

   and:

$$\mathbf{C}_g {}^{n+1}\mathbf{A}_g = {}^{n+1}\mathbf{A}_{g,\Gamma}. \tag{3.37}$$

2. a local analysis in the mesh Ω_ℓ^h defined in Figure 3.5, so that $\forall n \in \{0,\ldots,N-1\}$:

$$\mathbf{M}_\ell \left({}^{m+1}\mathbf{A}_\ell\right)^i = {}^{m+1}\mathbf{F}_\ell^{\text{ext}} - \left({}^{m+1}\mathbf{F}_\ell^{\text{int}}\right)^i \quad \forall m \in \{0,\ldots,M-1\} \tag{3.38}$$

   imposing the continuity of the local velocities with the global ones over the interface Γ_ℓ^h, so that $\forall n \in \{0,\ldots,N-1\}$:

$$\begin{aligned}\left({}^{m+1}\mathbf{V}_{\ell,\Gamma}\right)^i = \Pi_h^1 \Pi_t^3 \left({}^{n}\mathbf{V}_{g,\Gamma}, {}^{n}\mathbf{A}_{g,\Gamma}, \left({}^{n+1}\mathbf{V}_{g,\Gamma}\right)^i, \left({}^{n+1}\mathbf{A}_{g,\Gamma}\right)^i\right) \\ \forall m \in \{0,\ldots,M-1\}.\end{aligned} \tag{3.39}$$

3.5.1 Decomposition in pre-computation and correction phases

The resolution procedure of Equation (3.35) is decomposed in two phases:

1. the *pre-computation* phase: the global problem is solved using the coarse mesh and coarse time discretization everywhere; their output is computed only once using information available from the previous time step;

2. the *correction* phase: new estimates of the *correction forces* $\left({}^{n+1}\mathbf{P}\right)^i$ over the interface Γ_g^h are computed by repeated solutions of the local problem, involving the fine space and time discretizations.

The global decomposition in *pre-computation* and *correction* terms so reads:

$$\left({}^{n+1}\mathbf{A}_g\right)^i = {}^{n+1}\mathbf{A}_g^{\text{prec}} + \left({}^{n+1}\mathbf{A}_g^{\text{corr}}\right)^i \tag{3.40}$$

so that $\forall n \in \{0,\ldots,N-1\}$:

$$\begin{aligned}\mathbf{M}_g {}^{n+1}\mathbf{A}_g^{\text{prec}} &= {}^{n+1}\mathbf{F}_g^{\text{ext}} - {}^{n+1}\mathbf{F}_g^{\text{int}} && \text{in } \Omega_g^h|_{t^{n+1}} \\ \mathbf{M}_{g,\Gamma} \left({}^{n+1}\mathbf{A}_{g,\Gamma}^{\text{corr}}\right)^i &= \left({}^{n+1}\mathbf{P}\right)^i && \text{over } \Gamma_g^h|_{t^{n+1}}\end{aligned} \tag{3.41}$$

3.5.2 Steps of the algorithm

The algorithm to solve Equations (3.35) and (3.38) is divided in 3 main steps per each global time step:

1. global *pre-computation* phase in the overall mesh Ω_g^h, as in classical Finite Element analysis the solution is computed in function of external and internal forces:

$$\mathbf{M}_g {}^{n+1}\mathbf{A}_g^{\text{prec}} = {}^{n+1}\mathbf{F}_g^{\text{ext}} - {}^{n+1}\mathbf{F}_g^{\text{int}}; \tag{3.42}$$

2. global-local iterative *correction* phase over the global interface Γ_g^h, substituting the *global substitution* mesh Ω_{gs}^h by the *local* one Ω_ℓ^h and applying the *correction forces*:

    (a) local analysis in the refined mesh Ω_ℓ^h (initializations $i = 1$ and $\left({}^{n+1}\mathbf{A}_{g,\Gamma}^{\text{corr}}\right)^1 = \mathbf{0}$):

    $$\mathbf{M}_\ell \left({}^{m+1}\mathbf{A}_\ell\right)^i = {}^{m+1}\mathbf{F}_\ell^{\text{ext}} - \left({}^{m+1}\mathbf{F}_\ell^{\text{int}}\right)^i \quad \forall m \in \{0, \ldots, M-1\} \tag{3.43}$$

    with boundary conditions over the local interface Γ_ℓ^h as:

    $$\left({}^{m+1}\mathbf{V}_{\ell,\Gamma}\right)^i = \Pi_h^1 \Pi_t^3 \left({}^{n}\mathbf{V}_{g,\Gamma}, {}^{n}\mathbf{A}_{g,\Gamma}, \left({}^{n+1}\mathbf{V}_{g,\Gamma}\right)^i, \left({}^{n+1}\mathbf{A}_{g,\Gamma}\right)^i\right) \tag{3.44}$$

    and:

    $$\left({}^{n+1}\mathbf{A}_{g,\Gamma}\right)^i = {}^{n+1}\mathbf{A}_{g,\Gamma}^{\text{prec}} + \left({}^{n+1}\mathbf{A}_{g,\Gamma}^{\text{corr}}\right)^i; \tag{3.45}$$

    (b) global *correction* over the global interface Γ_g^h:

    $$\mathbf{M}_{g,\Gamma} \left({}^{n+1}\mathbf{A}_{g,\Gamma}^{\text{corr}}\right)^i = \left({}^{n+1}\mathbf{P}\right)^i \tag{3.46}$$

    with:

    $$\begin{aligned}\left({}^{n+1}\mathbf{P}\right)^i = \Pi_h^{1T} \left({}^{n+1}\mathbf{F}_{\ell,\Gamma}^{\text{ext}} - \mathbf{M}_{\ell,\Gamma} \left({}^{n+1}\mathbf{A}_{\ell,\Gamma}\right)^i - \left({}^{n+1}\mathbf{F}_{\ell,\Gamma}^{\text{int}}\right)^i\right) + \\ - {}^{n+1}\mathbf{F}_{gs,\Gamma}^{\text{ext}} + \mathbf{M}_{gs,\Gamma} \left({}^{n+1}\mathbf{A}_{g,\Gamma}\right)^{i-1} + {}^{n+1}\mathbf{F}_{gs,\Gamma}^{\text{int}};\end{aligned} \tag{3.47}$$

    (c) re-iteration for the sub-step (a) until the equilibrium is satisfied, i.e. the convergence measure e in Equation (3.22) is lower than a pre-fixed tolerance $\bar{e}$ and update of i to $i+1$;

3. re-adjustment of the global solution covering the gap between global and local analyses due to the tolerance and the iteration counter difference (the global solution belongs to the iteration $i-1$, though the local one belongs to the iteration i):

$$\mathbf{M}_{g,\Gamma} {}^{n+1}\mathbf{A}_{g,\Gamma}^{\text{resid}} = -\left({}^{n+1}\mathbf{R}_{gc,\Gamma}\right)^{\bar{i}} - \Pi_h^{1T} \left({}^{n+1}\mathbf{R}_{\ell,\Gamma}\right)^{\bar{i}} \tag{3.48}$$

and:

$${}^{n+1}\mathbf{A}_{g,\Gamma} = {}^{n+1}\mathbf{A}_{g,\Gamma}^{\text{prec}} + \left({}^{n+1}\mathbf{A}_{g,\Gamma}^{\text{corr}}\right)^{\bar{i}} + {}^{n+1}\mathbf{A}_{g,\Gamma}^{\text{resid}}. \tag{3.49}$$

It is important to note that ${}^{n+1}\mathbf{A}^{\mathrm{resid}}_{g,\Gamma}$ is a residual term proportional to the pre-fixed tolerance and ${}^{n+1}\mathbf{A}^{\mathrm{resid}}_{g,\Gamma} \ll {}^{n+1}\mathbf{A}^{\mathrm{corr}}_{g,\Gamma}$ if the tolerance is small enough.

Remark 9 *Satisfying Equations* (3.48) *and* (3.49) *at the step 5 of the algorithm, the solution only depends on the iteration $i-1$. Indeed, considering only the corrected accelerations and inverting the mass matrix:*

$$\left({}^{n+1}\mathbf{A}^{corr}_{g,\Gamma}\right)^i = \mathbf{M}^{-1}_{g,\Gamma}\left({}^{n+1}\mathbf{P}\right)^i = \mathbf{M}^{-1}_{g,\Gamma}\left(\Pi_h^{1T}\left({}^{n+1}\mathbf{R}_{\ell,\Gamma}\right)^i - \left({}^{n+1}\mathbf{R}_{gs,\Gamma}\right)^i\right) \tag{3.50}$$

where the term $\left({}^{n+1}\mathbf{R}_{\ell,\Gamma}\right)^i$ depends on the local accelerations $\left({}^{n+1}\mathbf{A}_{\ell,\Gamma}\right)^i$ at the iteration i, whereas the term $\left({}^{n+1}\mathbf{R}_{gs,\Gamma}\right)^i$ depends on the global accelerations $\left({}^{n+1}\mathbf{A}_{g,\Gamma}\right)^{i-1}$ at the iteration $i-1$. On the other hand the residual accelerations read:

$${}^{n+1}\mathbf{A}^{resid}_{g,\Gamma} = \mathbf{M}^{-1}_{g,\Gamma}\left(-\left({}^{n+1}\mathbf{R}_{gc,\Gamma}\right)^i - \Pi_h^{1T}\left({}^{n+1}\mathbf{R}_{\ell,\Gamma}\right)^i\right) \tag{3.51}$$

and the sum of corrected and residual accelerations becomes:

$$\begin{aligned}\left({}^{n+1}\mathbf{A}^{corr}_{g,\Gamma}\right)^i + {}^{n+1}\mathbf{A}^{resid}_{g,\Gamma} = \mathbf{M}^{-1}_{g,\Gamma}\left(\Pi_h^{1T}\left({}^{n+1}\mathbf{R}_{\ell,\Gamma}\right)^i - \left({}^{n+1}\mathbf{R}_{gs,\Gamma}\right)^i\right) + \\ -\mathbf{M}^{-1}_{g,\Gamma}\left(\left({}^{n+1}\mathbf{R}_{gc,\Gamma}\right)^i + \Pi_h^{1T}\left({}^{n+1}\mathbf{R}_{\ell,\Gamma}\right)^i\right) = \\ = -\mathbf{M}^{-1}_{g,\Gamma}\left(\Pi_h^{1T}\left({}^{n+1}\mathbf{R}_{gs,\Gamma}\right)^i + \left({}^{n+1}\mathbf{R}_{gc,\Gamma}\right)^i\right).\end{aligned} \tag{3.52}$$

The flow chart in Algorithm 9 summarizes the procedure steps above.

```
while $t_g < t_{end}$ do
    global pre-computation: $\mathbf{M}_g{}^{n+1}\mathbf{A}_g^{\text{prec}} = {}^{n+1}\mathbf{F}_g^{\text{ext}} - {}^{n+1}\mathbf{F}_g^{\text{int}}$;
    if Substitution is activated then
        initialize $\left({}^{n+1}\mathbf{A}_{g,\Gamma}\right)^0 \leftarrow {}^{n+1}\mathbf{A}_{g,\Gamma}^{\text{prec}}$;
        while $e > \bar{e}$ do
            back in time $t_\ell \leftarrow t_g$;
            while $t_\ell < t_g + \Delta t_g$ do
                local computation: $\mathbf{M}_\ell\left({}^{m+1}\mathbf{A}_\ell\right)^i = {}^{m+1}\mathbf{F}_\ell^{\text{ext}} - \left({}^{m+1}\mathbf{F}_\ell^{\text{int}}\right)^i$;
                with:
                ${}^{m+1}\mathbf{V}_{\ell,\Gamma} \leftarrow \Pi_h^1\Pi_t^3\left({}^{n}\mathbf{V}_{g,\Gamma}, {}^{n}\mathbf{A}_{g,\Gamma}, \left({}^{n+1}\mathbf{V}_{g,\Gamma}\right)^i, \left({}^{n+1}\mathbf{A}_{g,\Gamma}\right)^i\right)$;
                update $t_\ell \leftarrow t_\ell + \Delta t_\ell$;
            end
            global correction: $\mathbf{M}_{g,\Gamma}\left({}^{n+1}\mathbf{A}_{g,\Gamma}^{\text{corr}}\right)^i = \left({}^{n+1}\mathbf{P}\right)^i$;
            update $\left({}^{n+1}\mathbf{A}_{g,\Gamma}\right)^i \leftarrow {}^{n+1}\mathbf{A}_{g,\Gamma}^{\text{prec}} + \left({}^{n+1}\mathbf{A}_{g,\Gamma}^{\text{corr}}\right)^i$;
            update $i \leftarrow i + 1$;
            compute $e$;
        end
        global stabilization: $\mathbf{M}_{g,\Gamma}{}^{n+1}\mathbf{A}_{g,\Gamma}^{\text{resid}} = -\left({}^{n+1}\mathbf{R}_{gc}\right)^{\bar{i}} - \Pi_h^{1T}\left({}^{n+1}\mathbf{R}_\ell\right)^{\bar{i}}$;
        update ${}^{n+1}\mathbf{A}_{g,\Gamma} \leftarrow {}^{n+1}\mathbf{A}_{g,\Gamma}^{\text{prec}} + \left({}^{n+1}\mathbf{A}_{g,\Gamma}^{\text{corr}}\right)^{\bar{i}} + {}^{n+1}\mathbf{A}_{g,\Gamma}^{\text{resid}}$;
    end
    update $t_g \leftarrow t_g + \Delta t_g$;
end
```

Algorithm 9: Substitution iterative process.

3.6 Application and results analysis

The initial-boundary value problem introduced in System (1.1) is applied to solve the *reference problem* described in Figure 3.13 via the Substitution method.

The data of the problem are as follows:

- density $\rho = 7800\ kg/m^3$;
- Young's modulus $E = 210 \cdot 10^9\ Pa$;
- Poisson's ratio $\nu = 0.3$;
- length $L = 30\ m$;
- height $H = 10\ m$;

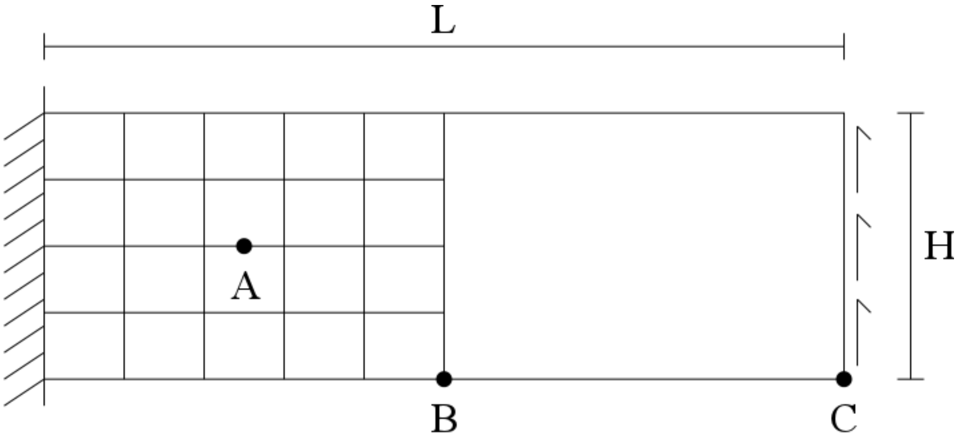


Figure 3.13: Bondary conditions, loads, sizes and monitored locations of the problem.

- initial time $t_{\text{in}} = 0\ s$;
- final time $t_{\text{end}} = 0.5\ s$;
- uniform shear load $\overline{f} = 3 \cdot 10^6\ Pa$ constant in time.

The critical time step bound Δt_{cr} is estimated with Equation (1.20) in Section 3.1 of Chapter 1, leading to $\Delta t_{\text{cr}} \simeq 13 \cdot 10^{-4}\ s$ for the global analysis. The solution history is observed at three representative locations of the domain: location A internal to the local region Ω_ℓ, location B over the interface Γ between local and *global complementary* regions and location C at the extremity of the structure (see Figure 3.13).

The Substitution method is validated by comparison with the Domain Decomposition *GCbis* algorithm.

3.6.1 Comparison with Domain Decomposition GCbis algorithm and influence of the time step size

The analysis is here carried out using pre-fixed time steps close to the critical bound Δt_{cr}, so that:

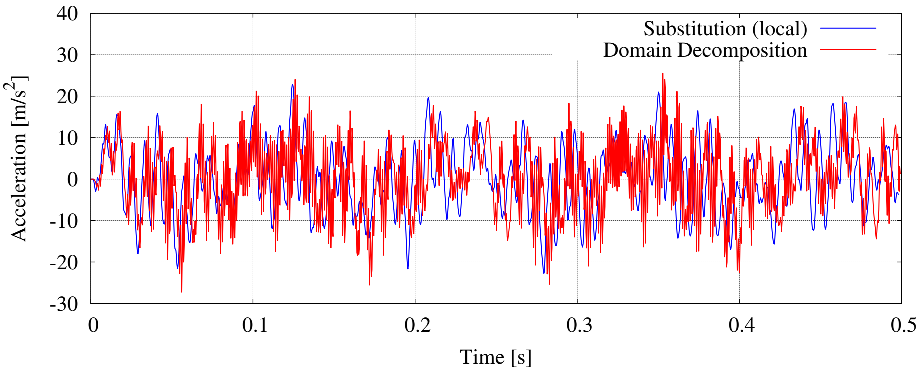
- $\Delta t_g = 12 \cdot 10^{-4}\ s$ for the global analysis and the coarse subdomain,
- $\Delta t_\ell = \Delta t_g / M = 3 \cdot 10^{-4}\ s$ for the local analysis and the fine subdomain,

with the time step ratio M equal to the global-local mesh ratio $h_{g/\ell}$, such that $M = h_{g/\ell} = 4$.

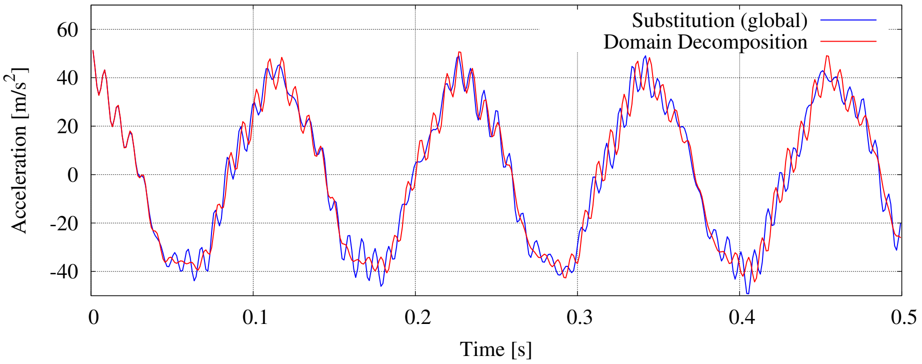
A convergence tolerance $\overline{e} = 1\%$ is considered for the comparison.

Figures 3.14 show the results in terms of vertical accelerations at the locations A and C. The curves perfectly match at the beginning of the analysis until $t = 0.05\ s$, after which some discrepancies become visible. Nevertheless, it is difficult to assess which curve denotes the most accurate solution.

Refining the time steps and keeping same time step ratio M and global-local mesh ratio $h_{g/\ell}$ such that $M = h_{g/\ell} = 4$, a second analysis is run with:



(a) location A

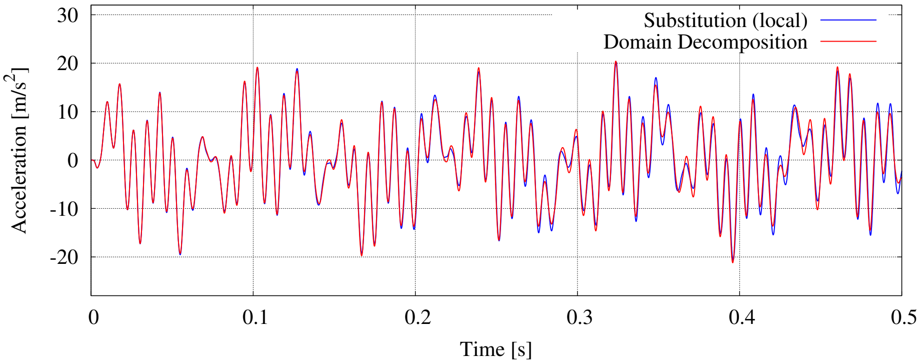


(b) location C

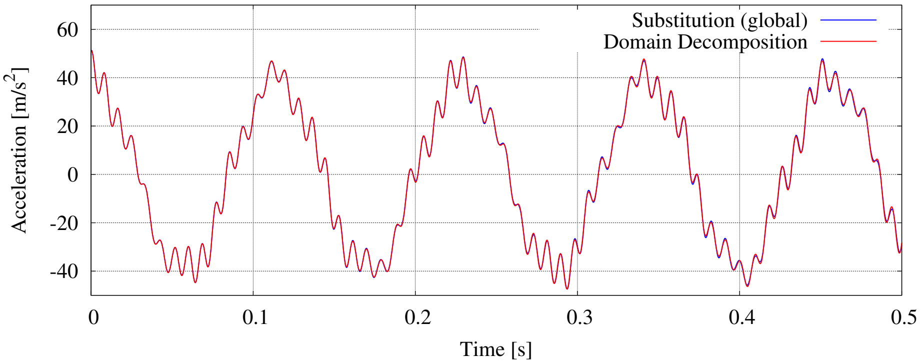
Figure 3.14: Comparison between Substitution and Domain Decomposition (GCbis) in terms of vertical accelerations.

- $\Delta t_g = 4 \cdot 10^{-4}$ s for the global analysis and the coarse subdomain;
- $\Delta t_\ell = 1 \cdot 10^{-4}$ s for the local analysis and the fine subdomain.

Figures 3.15 show the comparison in terms of vertical accelerations and the curves of Substitution and Domain Decomposition algorithm match during the whole analysis. This demonstrates that, refining the time discretization, the solutions of Substitution and Domain Decomposition converge to the same exact solution.



(a) location A

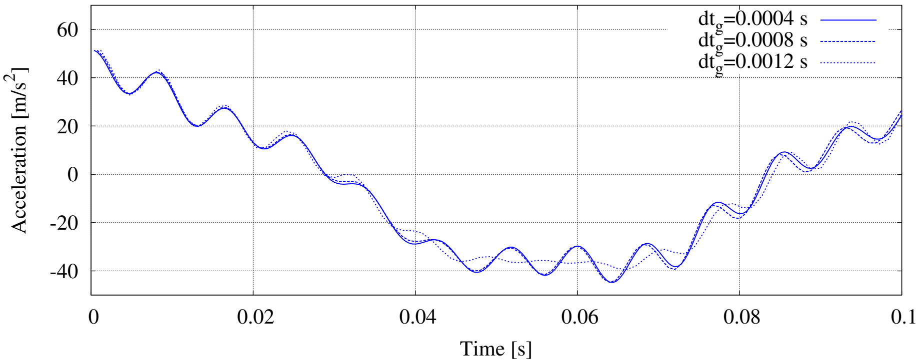


(b) location C

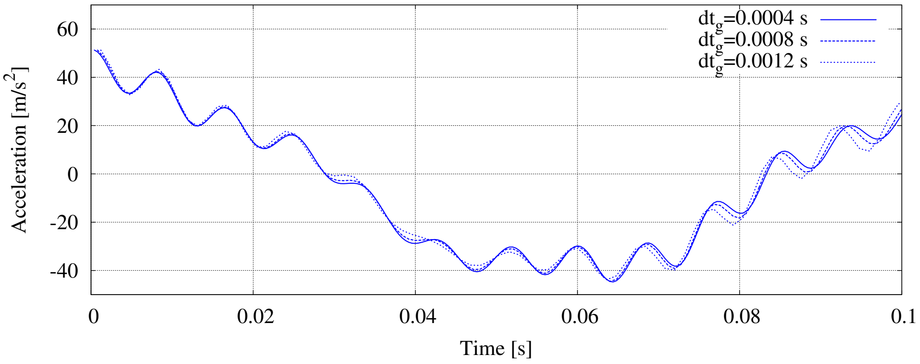
Figure 3.15: Comparison between Substitution and Domain Decomposition (GCbis) in terms of accelerations refining both the time steps.

In particular, Figures 3.16 compare the vertical accelerations of the two methods at the location C at the beginning of the analysis until $t = 0.1$ s, considering the refinement of the global time step with $\Delta t_g = 0.0012$ s, 0.0008 s and 0.0004 s and maintaining $M = h_{g/\ell} = 4$. Figure 3.16(a) shows the convergence of the Domain Decomposition *GCbis* algorithm toward the exact solution, whereas Figure 3.16(b) shows the convergence of the Substitution. The results denote that the Substitution is more accurate with a coarse time step but cumulates a larger phase difference during the analysis.

Further numerical studies should be conducted to assess the accuracy of the Substitution method in comparison with Domain Decomposition methods.



(a) Domain Decomposition (GCbis)



(b) Substitution

Figure 3.16: Vertical acceleration at the location C at the beginning of the analysis refining the time steps.

3.6.2 Convergence of the iterative procedure

As introduced in Section 3.3, a fixed-point iterative algorithm is employed to search for the equilibrium between the global and local analyses. The convergence measure of the iterative process is chosen to be the equilibrium-based convergence measure e of Equation (3.22).

The pre-fixed tolerance is reduced to $\bar{e} = 0.01\%$ for this study.

Figure 3.17 shows the convergence rate of the iterative procedure. The convergence measure is plotted in a logarithmic scale over the iterations at three representative times of the analysis $t_1 = t_{\text{end}}/3$, $t_2 = t_{\text{end}}/2$ and $t_3 = 2t_{\text{end}}/3$, extending the analysis to $t_{\text{end}} = 2.5\ s$. The results show a convergence rate almost constant during the iterative process and during the analysis.

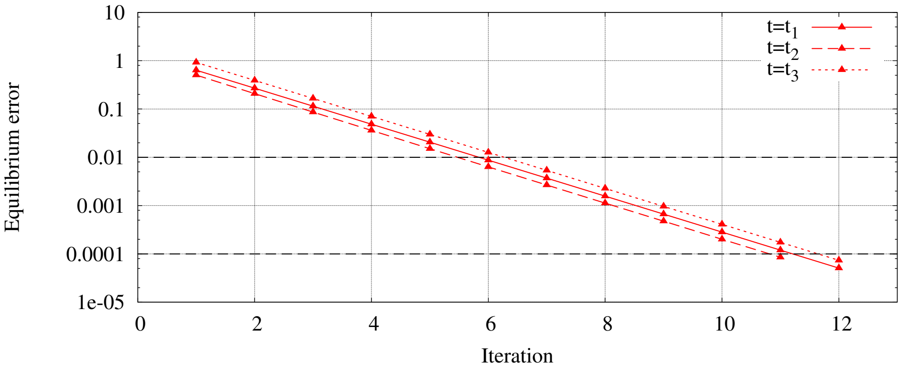


Figure 3.17: Convergence rate with $\bar{e} = 0.01\%$ at three time instants: $t_i = \frac{1+i}{6} t_{\text{end}}$ with $i = 1,2,3$ and $t_{\text{end}} = 2.5\ s$.

Figure 3.18 shows the required number of iterations to get an acceptable error $e < \bar{e} = 1\%$ in a common plot with the vertical displacements at the location B over the interface Γ_g^h. The number of iterations varies between 4 and 8 throughout the duration of the analysis and follows the trend of the displacement solution, because the convergence measure e is normalized by a term independent of time, while the reaction forces depend on the configuration of the domain.

Figure 3.19 shows the solution in terms of vertical accelerations at the location B over the interface Γ_g^h at the different iterations in a representative time interval between $t = 0.214\ s$ and $t = 0.222\ s$. The red markers denote the solution at the iterations and the solid blue line indicates the solution at convergence $e < \bar{e}$. The acceleration solution at the iteration 5 is already close to the final outcome.

3.6.3 Stability study

The energy method, introduced in Section 4.2.4 of Chapter 1 for Domain Decomposition couplings, is used here to test the stability of the Substitution method. In FETI Domain Decomposition, Lagrange multipliers are applied over the interfaces to ensure kinematic

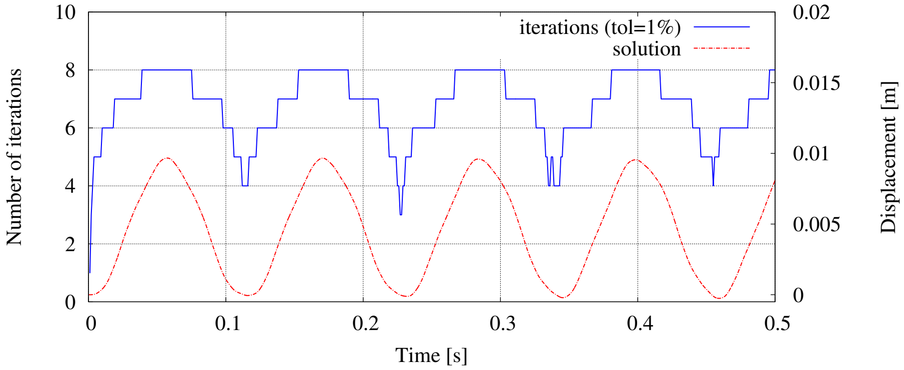


Figure 3.18: Displacements and number of iterations with $\bar{e} = 1\%$ during the simulation.

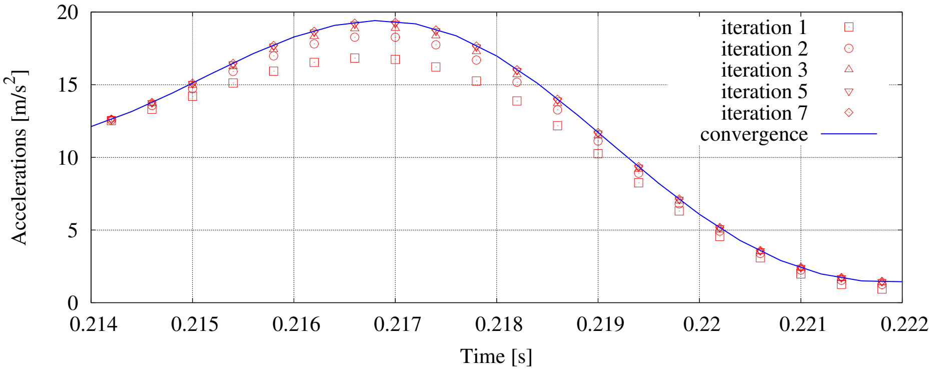


Figure 3.19: Vertical accelerations at the location B over the iterations.

continuity and equilibrium, without iterations. Conversely, in Substitution, the kinematic continuity is applied through local boundary conditions and the equilibrium is searched through an iterative process. Whereas the stability indicator was connected to Lagrange multipliers in Domain Decomposition, here it is expressed in function of the reaction forces over the global interface due to the local boundary conditions and the global corrections, so that:

$$ {}^{n}E_{\Gamma} = \frac{1}{\Delta t_{\ell}} \Pi_h^{1T} [\![{}^{n}\mathbf{V}_{\ell,\Gamma}]\!] [\![{}^{n}\mathbf{R}_{\ell,\Gamma}]\!] + \frac{1}{\Delta t_g} [{}^{n}\mathbf{V}_{gc,\Gamma}] [{}^{n}\mathbf{R}_{gc,\Gamma}] . \tag{3.53} $$

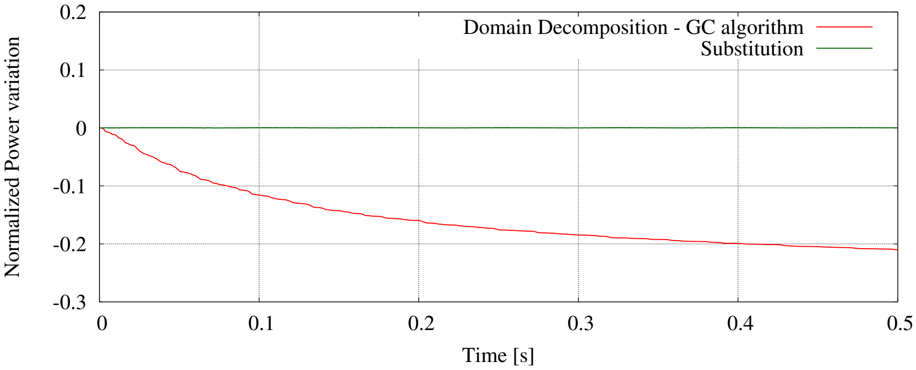
where, as introduced in Section 4.2.4 of Chapter 1, $[\square] = {}^{n+1}\square - {}^{n}\square$ is associated to the global time step and $[\![\square]\!] = \sum_{m=0}^{M-1} \left({}^{m+1}\square - {}^{m}\square \right)$ is associated to the local time stepping.

As for Domain Decomposition in Section 2.3 of Chapter 2, the stability indicator is normalized by the external force term cumulated at the end of the analysis.

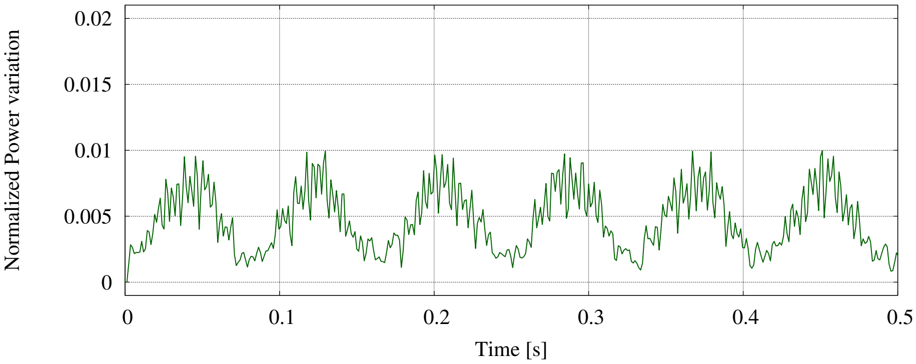
A convergence tolerance $\bar{e} = 1\%$ is considered for this case of study.

The results in terms of stability indicators are shown in Figure 3.20: Figure 3.20(a) shows the comparison between stability indicators of Domain Decomposition with *GC* algorithm and Substitution. Where the red line, indicating the numerical dissipation of

Domain Decomposition (GC), reaches absolute values around 20% that tend to grow, the green curve associated to Substitution is approximatively null. Figure 3.20(b) focuses on the stability indicator of the Substitution method and shows that the values steadily oscillate around 0.5% but are not dissipative. It is important to note that such oscillations are not in phase with the global solution. Further studies are necessary to assess the oscillatory source of the stability indicator.



(a) Comparison between Substitution and Domain Decomposition (GC)



(b) Substitution

Figure 3.20: Stability indicators.

The same analysis is run longer until the time $t_{\text{end}} = 10\ s$, with more than 8000 time steps. Figure 3.21 shows the stability indicator always oscillating around negligible values.

Further analyses are carried out in order to study the variation of the stability indicator in function of the discretization parameters as the time steps Δt_g and Δt_ℓ, their ratio M and the global-local mesh ratio $h_{g/\ell}$.

Firstly, the meshes and the time step ratio M are conserved and the time steps Δt_g and Δt_ℓ are refined, so that three analyses are run with:

1. $\Delta t_g = 12 \cdot 10^{-4}\ s$ and $\Delta t_\ell = \Delta t_g / M = 3 \cdot 10^{-4}\ s$ with $M = 4$;
2. $\Delta t_g = 8 \cdot 10^{-4}\ s$ and $\Delta t_\ell = \Delta t_g / M = 2 \cdot 10^{-4}\ s$ with $M = 4$;

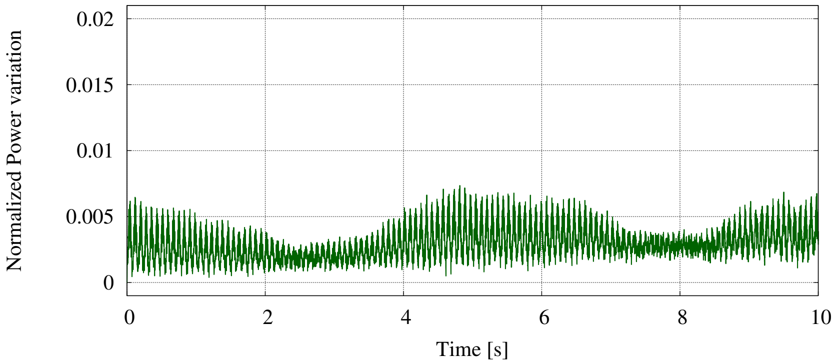


Figure 3.21: Stability indicator with longer analysis with $t_{\text{end}} = 10\ s$.

3. $\Delta t_g = 4 \cdot 10^{-4}\ s$ and $\Delta t_\ell = \Delta t_g / M = 1 \cdot 10^{-4}\ s$ with $M = 4$.

Figure 3.22 shows the decrease of the stability indicator values proportional to the decrease of the time steps. In every analysis, the values steadily oscillate around a given value, that is decreasing in function of the time steps.

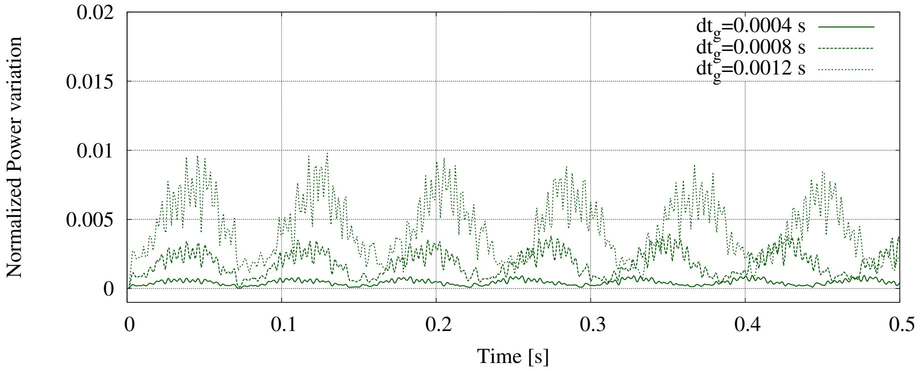


Figure 3.22: Stability indicator comparisons decreasing the time steps Δt_g and Δt_ℓ and fixing the time step ratio $M = 4$.

Then, as illustrated in Figure 3.23, the increase of the time step ratio M is considered keeping the same global and local meshes, so that three analyses are run with:

1. $\Delta t_g = 12 \cdot 10^{-4}\ s$ and $\Delta t_\ell = \Delta t_g / M = 3 \cdot 10^{-4}\ s$ with $M = 4$;
2. $\Delta t_g = 12 \cdot 10^{-4}\ s$ and $\Delta t_\ell = \Delta t_g / M = 1.5 \cdot 10^{-4}\ s$ with $M = 8$;
3. $\Delta t_g = 12 \cdot 10^{-4}\ s$ and $\Delta t_\ell = \Delta t_g / M = 0.75 \cdot 10^{-4}\ s$ with $M = 16$.

Figure 3.24 shows that the stability indicators of the three solutions follow approximatively the same paths and the values are nearly the same. So, the decrease of the local time step Δt_ℓ alone does not influence the stability of the coupling.

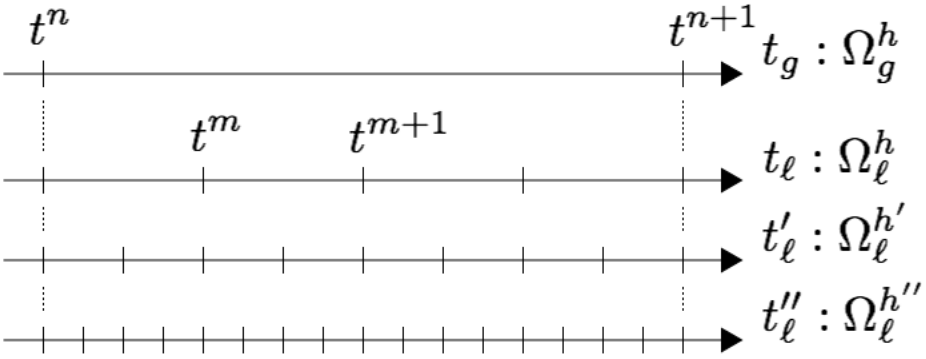


Figure 3.23: Refinement of the local time step Δt_ℓ, fixing the global one Δt_g.

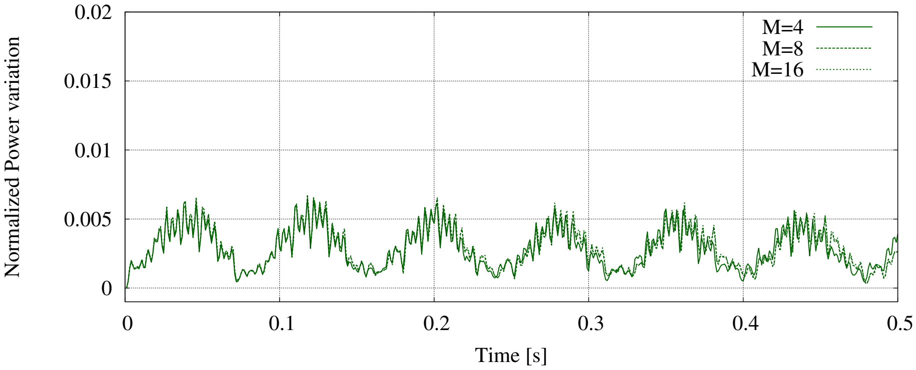


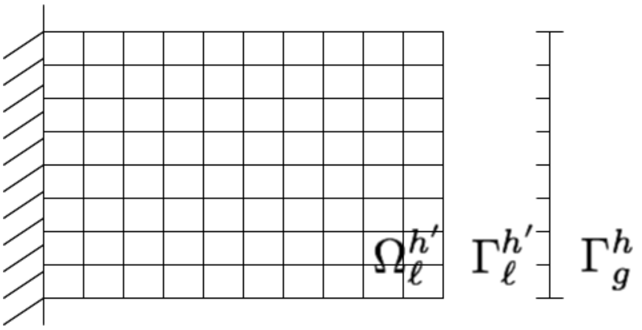
Figure 3.24: Stability indicator comparisons increasing the time step ratio M, fixing the global time step Δt_g.

At last, the meshes illustrated in Figure 3.25 are considered in order to test the influence of the global-local mesh size ratio $h_{g/\ell}$ onto the stability indicator. In addition to the analysis with $M = h_{g/\ell} = 4$, two additional analyses with refinement are performed. A low global time step is chosen. The three analyses are so run with:

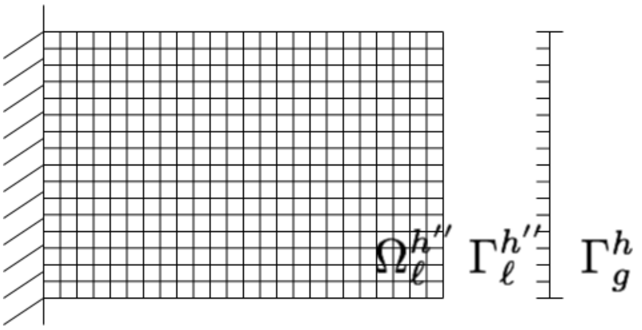
1. $\Delta t_g = 4 \cdot 10^{-4}\ s$ and $\Delta t_\ell = \Delta t_g / M = 1.0 \cdot 10^{-4}\ s$ with $M = h_{g/\ell} = 4$;

2. $\Delta t_g = 4 \cdot 10^{-4}\ s$ and $\Delta t_\ell = \Delta t_g / M = 0.5 \cdot 10^{-4}\ s$ with $M = h_{g/\ell} = 8$ and the local mesh as in Figure 3.25(a);

3. $\Delta t_g = 4 \cdot 10^{-4}\ s$ and $\Delta t_\ell = \Delta t_g / M = 0.25 \cdot 10^{-4}\ s$ with $M = h_{g/\ell} = 16$ and the local mesh as in Figure 3.25(b).



(a) global-local mesh size ratio $h_{g/\ell} = 8$



(b) global-local mesh size ratio $h_{g/\ell} = 16$

Figure 3.25: Mesh refinement of the local analysis.

Figure 3.26 shows the results related to the mesh refinement. Once again, the stability indicator oscillates around the similar values, though a slightly difference phase is remarked. Furthermore, the sensitivity of the stability indicator is remarked with such refinement. Indeed, one may observe that increasing the global time step Δt_g the curves are not easily comparable. Further studies are necessary to define the stability of the algorithm in function of the global-local interface refinement $h_{g/\ell}$.

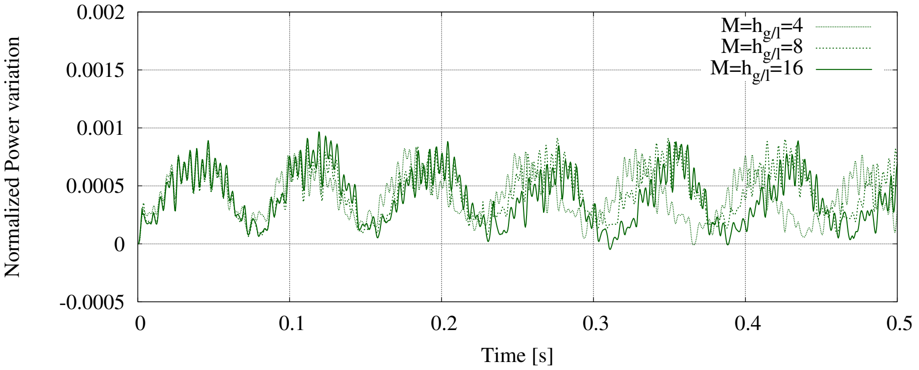


Figure 3.26: Stability indicator comparing different time step ratios M and global-local mesh ratios $h_{g/\ell}$.

4 Energy-preserving Substitution method

By analogy with the Domain Decomposition evolution, an alternative energy-preserving version of the Substitution method is here proposed. A new formulation with Lagrange multipliers inspired to the remarks in [Prakash and Hjelmstad, 2004] is considered. The continuity of velocities is enforced only at the global level and the Lagrange multipliers are linearly interpolated in time over the local time stepping.

4.1 Definition of the new interface constraint

The local momentum balance reads:

$$\mathbf{M}_\ell \left({}^{m+1}\mathbf{A}_\ell\right)^i = {}^{m+1}\mathbf{F}_\ell^{\text{ext}} - {}^{m+1}\mathbf{F}_\ell^{\text{int}} + \mathbf{C}_\ell^T \Pi_t^1 \left({}^{n}\boldsymbol{\Lambda}, \left({}^{n+1}\boldsymbol{\Lambda}\right)^i\right) \tag{3.54}$$

where $\mathbf{C}_\ell$ is the Boolean unitary matrix that extracts nodal values from the local domain to the local interface, so that:

$$\mathbf{C}_{\ell,(\dim\Gamma_\ell^h \times \dim\Omega_\ell^h)} : \Omega_\ell^h \mapsto \Gamma_\ell^h \tag{3.55}$$

and:

$$\mathbf{C}_\ell {}^{n+1}\mathbf{A}_\ell = {}^{n+1}\mathbf{A}_{\ell,\Gamma} \tag{3.56}$$

and the constraint of velocities at each global time step reads:

$$\left({}^{n+1}\mathbf{V}_{\ell,\Gamma}\right)^i = \Pi_h^1 \left({}^{n+1}\mathbf{V}_{g,\Gamma}\right)^i. \tag{3.57}$$

As in the case of the energy-preserving multi-time-step FETI approach, the formulation utilizes a monolithic matrix notation comprehending displacements, velocities and accelerations in the local time stepping between the global times t^n and t^{n+1}.

Defining the local matrices:

$$\mathbb{M}_\ell = \begin{bmatrix} \mathbf{M}_\ell & \mathbf{0} & \mathbf{0} \\ -\mathbf{I}\frac{\Delta t_\ell}{2} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} \end{bmatrix} \quad \mathbb{N}_\ell = \begin{bmatrix} \mathbb{K}_\ell \frac{\Delta t_\ell^2}{2} & \mathbb{K}_\ell \Delta t_\ell & \mathbb{K}_\ell \\ -\mathbf{I}\frac{\Delta t_\ell}{2} & -\mathbf{I} & \mathbf{0} \\ -\mathbf{I}\frac{\Delta t_\ell^2}{2} & -\mathbf{I}\Delta t_\ell & -\mathbf{I} \end{bmatrix}$$

$$\mathbf{X}_\ell = \begin{Bmatrix} \mathbf{A}_\ell \\ \mathbf{V}_\ell \\ \mathbf{U}_\ell \end{Bmatrix} \quad \mathbb{F}_\ell^{\text{ext}} = \begin{Bmatrix} \mathbf{F}_\ell^{\text{ext}} \\ \mathbf{0} \\ \mathbf{0} \end{Bmatrix} \quad \mathbb{C}_\ell = \begin{bmatrix} \mathbf{C}_\ell^T \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} \quad \mathbb{B}_\ell = \begin{bmatrix} \mathbf{0} & \mathbf{C}_\ell & \mathbf{0} \end{bmatrix}$$

where $\mathbb{K}_\ell$ is the local stiffness matrix. Equations (3.54) and (3.57) at each global time step are rewritten in a unified way as:

$$\begin{bmatrix} \mathbb{M}_\ell & & & & \frac{1}{M}\mathbb{C}_\ell \\ \mathbb{N}_\ell & \mathbb{M}_\ell & & & \frac{2}{M}\mathbb{C}_\ell \\ & \ddots & \ddots & & \vdots \\ & & \mathbb{N}_\ell & \mathbb{M}_\ell & \mathbb{C}_\ell \\ & & & \mathbb{B}_\ell & \end{bmatrix} \begin{Bmatrix} {}^{1}\mathbf{X}_\ell \\ {}^{2}\mathbf{X}_\ell \\ \vdots \\ {}^{M}\mathbf{X}_\ell \\ {}^{n+1}\mathbf{\Lambda} \end{Bmatrix} = \begin{Bmatrix} {}^{1}\mathbb{F}_\ell^{\text{ext}} - \mathbb{N}_\ell{}^{0}\mathbf{X}_\ell + \frac{M-1}{M}\mathbb{C}_\ell{}^{n}\mathbf{\Lambda} \\ {}^{2}\mathbb{F}_\ell^{\text{ext}} + \frac{M-2}{M}\mathbb{C}_\ell{}^{n}\mathbf{\Lambda} \\ \vdots \\ {}^{M}\mathbb{F}_\ell^{\text{ext}} \\ \Pi_h^1\left({}^{n+1}\mathbf{V}_{g,\Gamma}\right)^i \end{Bmatrix} \tag{3.58}$$

that becomes:

$$\begin{bmatrix} \mathsf{M}_\ell & \mathsf{C}_\ell \\ \mathsf{B}_\ell & \mathbf{0} \end{bmatrix} \begin{Bmatrix} {}^{n+1}\mathsf{X}_\ell \\ {}^{n+1}\mathbf{\Lambda} \end{Bmatrix} = \begin{Bmatrix} \mathsf{F}_\ell \\ \mathbf{0} \end{Bmatrix} \tag{3.59}$$

taking M_ℓ as a lower triangular matrix composed of matrices $\mathbb{M}_\ell$ and $\mathbb{N}_\ell$, C_ℓ as a matrix containing the linear interpolation of matrices $\mathbb{C}_\ell$, B_ℓ as a Boolean matrix which consists of matrix $\mathbb{B}_\ell$, X_ℓ is the displacement-velocity-acceleration vector and F_ℓ is the force vector, containing also the first term of the Lagrange multipliers' linear interpolation.

As for Domain Decomposition, a *free-link* separation approach is used to decompose the variable vector X_ℓ into $\mathsf{X}_\ell^{\text{free}} + \mathsf{X}_\ell^{\text{link}}$. Lagrange multipliers are so defined as:

$$\mathsf{M}_\ell{}^{n+1}\mathsf{X}_\ell^{\text{link}} = \mathsf{C}_\ell{}^{n+1}\mathbf{\Lambda}. \tag{3.60}$$

System (3.59) can be solved in three steps:

1. resolution of the *free* problem:

$$\mathsf{M}_\ell{}^{n+1}\mathsf{X}_\ell^{\text{free}} = \mathsf{F}_\ell; \tag{3.61}$$

2. calculation of the Lagrange multipliers with:

$$\mathsf{M}_\ell \mathbb{Y} = \mathsf{C}_\ell \tag{3.62}$$

and

$$\mathbb{H}_\ell{}^{n+1}\mathbf{\Lambda} = \mathsf{B}_\ell{}^{n+1}\mathsf{X}_\ell^{\text{free}} \tag{3.63}$$

where $\mathbb{H}_\ell = \mathsf{B}_\ell \mathbb{Y}$ is the condensation matrix and:

$$\mathsf{B}_\ell{}^{n+1}\mathsf{X}_\ell^{\text{free}} = \mathbf{C}_\ell{}^{M}\mathbf{V}_\ell^{\text{free}}; \tag{3.64}$$

3. resolution of the *link* problem:

$$ {}^{n+1}\mathsf{X}_{\text{link}} = \mathbb{Y}\,{}^{n+1}\boldsymbol{\Lambda}. \tag{3.65} $$

```
resolution of $\mathsf{M}_\ell \mathbb{Y}_\ell = \mathsf{C}_\ell$;
computation of $\mathbb{H}_\ell = \mathsf{B}_\ell \mathbb{Y}_\ell$;
while $t_g < t_{end}$ do
    global pre-computation: $\mathbf{M}_g {}^{n+1}\mathbf{A}_g^{\text{prec}} = {}^{n+1}\mathbf{F}_g^{\text{ext}} - {}^{n+1}\mathbf{F}_g^{\text{int}}$;
    if Substitution is activated then
        initialize $\left({}^{n+1}\mathbf{A}_{g,\Gamma}\right)^0 \leftarrow {}^{n+1}\mathbf{A}_{g,\Gamma}^{\text{prec}}$;
        while $e > \bar{e}$ do
            back in time $t_\ell \leftarrow t_g$;
            while $t_\ell^{free} < t_g + \Delta t_g$ do
                local free computation:
                $\mathbf{M}_\ell \left({}^{m+1}\mathbf{A}_\ell^{\text{free}}\right)^i = {}^{m+1}\mathbf{F}_\ell^{\text{ext}} - \left({}^{m+1}\mathbf{F}_\ell^{\text{int,free}}\right)^i + \mathbf{C}_\ell \left(1 - \frac{m}{M}\right) {}^{n}\boldsymbol{\Lambda}$;
                update $t_\ell^{\text{free}} \leftarrow t_\ell^{\text{free}} + \Delta t_\ell$;
            end
            interface constraint: $\mathbb{H}_\ell \left({}^{n+1}\boldsymbol{\Lambda}\right)^i = \left({}^{n+1}\mathbf{V}_{g,\Gamma}\right)^i - {}^{n+1}\mathbf{V}_{\ell,\Gamma}^{\text{free}}$;
            while $t_\ell^{link} < t_g + \Delta t_g$ do
                local link computation:
                $\mathbf{M}_\ell \left({}^{m+1}\mathbf{A}_\ell^{\text{link}}\right)^i = -\left({}^{m+1}\mathbf{F}_\ell^{\text{int,free}}\right)^i + \mathbf{C}_\ell \frac{m}{M} \left({}^{n+1}\boldsymbol{\Lambda}\right)^i$;
                update $t_\ell^{\text{link}} \leftarrow t_\ell^{\text{link}} + \Delta t_\ell$;
            end
            global correction: $\mathbf{M}_{g,\Gamma} \left({}^{n+1}\mathbf{A}_{g,\Gamma}^{\text{corr}}\right)^i = \left({}^{n+1}\mathbf{P}\right)^i$;
            update $\left({}^{n+1}\mathbf{A}_{g,\Gamma}\right)^i \leftarrow {}^{n+1}\mathbf{A}_{g,\Gamma}^{\text{prec}} + \left({}^{n+1}\mathbf{A}_{g,\Gamma}^{\text{corr}}\right)^i$;
            update $i \leftarrow i + 1$;
            compute $e$;
        end
        global stabilization: $\mathbf{M}_{g,\Gamma} {}^{n+1}\mathbf{A}_{g,\Gamma}^{\text{resid}} = -\left({}^{n+1}\mathbf{R}_{gc}\right)^{\bar{i}} - \Pi_h^{1T} \left({}^{n+1}\mathbf{R}_\ell\right)^{\bar{i}}$;
    end
    update ${}^{n+1}\mathbf{A}_{g,\Gamma} \leftarrow {}^{n+1}\mathbf{A}_{g,\Gamma}^{\text{prec}} + \left({}^{n+1}\mathbf{A}_{g,\Gamma}^{\text{corr}}\right)^{\bar{i}} + {}^{n+1}\mathbf{A}_{g,\Gamma}^{\text{resid}}$;
    update $t_g \leftarrow t_g + \Delta t_g$;
end
```

Algorithm 10: Energy-preserving Substitution algorithm.

All the variables of displacement, velocity and acceleration are decomposed in *free* variables and *link* ones. Involving also displacements, the condensation matrix $\mathbb{H}_\ell$ is composed of non-zero square blocks, also resulting from the stiffness matrix $\mathbb{K}_\ell$ contribution. Furthermore, the local analysis is computed in two distinct time loops t_ℓ^{free} and t_ℓ^{link}.

The steps described above are summarized in Algorithm 10.

4.2 Application and results analysis

The same problem analyzed in Section 3.6 is here considered.

4.2.1 Comparison with Domain Decomposition GCbis algorithm

For the comparison between energy-preserving Domain Decomposition and Substitution, the analysis is carried out using time steps close to the critical estimation Δt_{cr}, such that:

- $\Delta t_g = 12 \cdot 10^{-4}\ s$ for the global analysis and the coarse subdomain,
- $\Delta t_\ell = \Delta t_g / M = 3 \cdot 10^{-4}\ s$ for the local analysis and the fine subdomain,

with $M = h_{g/\ell} = 4$. A pre-fixed tolerance $\bar{e} = 1\%$ is used.

Figures 3.27 show the vertical accelerations at the locations A and C. Even with such coarse time discretization the solutions perfectly match throughout the analysis. Indeed, the very same energy-preserving formulation is employed for the velocity constraint definition.

4.2.2 Stability study

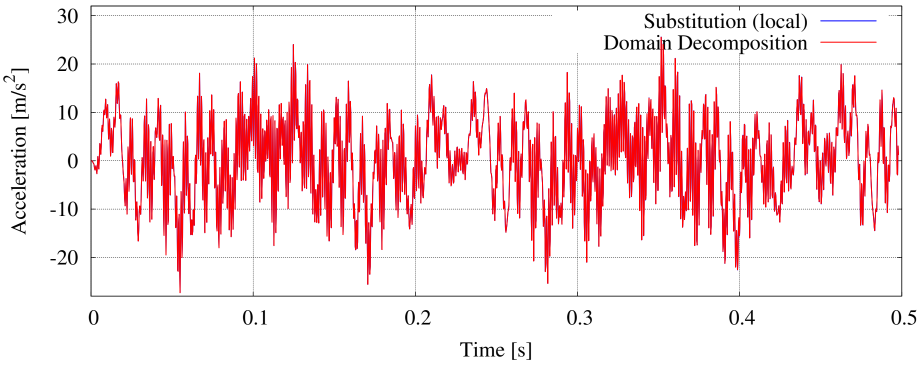
The energy method is used to verify the stability of the coupling scheme. The results shown in Figure 3.28 denote a null stability indicator, considering an analysis with $t_{\text{end}} = 10\ s$ and so more than $8,000$ global time steps. The coupling is shown to be perfectly conservative, considering the machine precision for running the analysis.

4.3 Conclusions

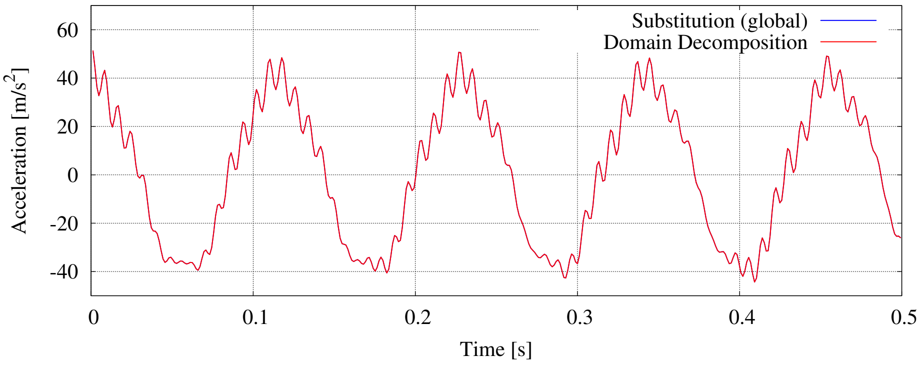
Although perfectly conservative, the energy-preserving Substitution method is not considered for further developments. Indeed, as for the energy-preserving Domain Decomposition methods, the extension of such reformulation to nonlinearities and adaptive time step could be costly and intrusive, because in that case the condensation matrix $\mathbb{H}_\ell$ should be re-assembled at each global iteration.

5 Example of simplified impact

The study of different geometries and test cases can usually evidence additional issues or show important characteristics. The impact test case described in Figure 3.29 is analyzed: for this purpose, a cylindrical body of mass $\mathcal{M}$ and velocity v_{imp} impacts a simply



(a) location A



(b) location C

Figure 3.27: Comparison between energy-preserving Substitution and Domain Decomposition (GCbis) in terms of vertical accelerations.

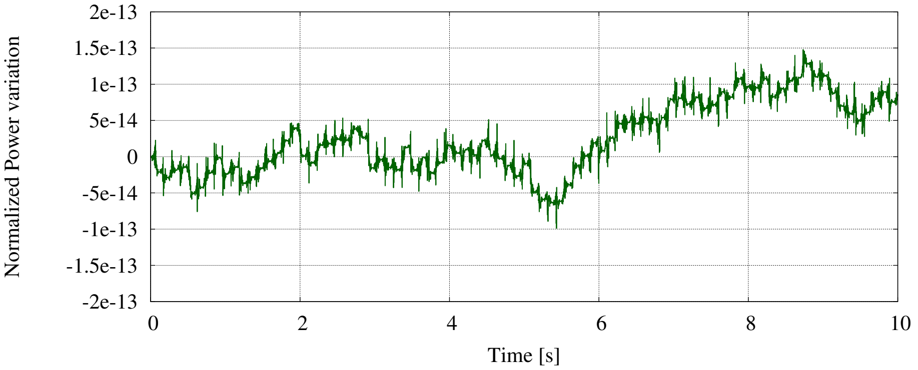


Figure 3.28: Stability indicator with $t_{\text{end}} = 10\ s$.

supported flat panel. The assumption of plane strains is taken in order to reduce the cylindrical bending problem to a 2-dimensional one. Kinematic quantities are observed during the analysis at the locations A and B.

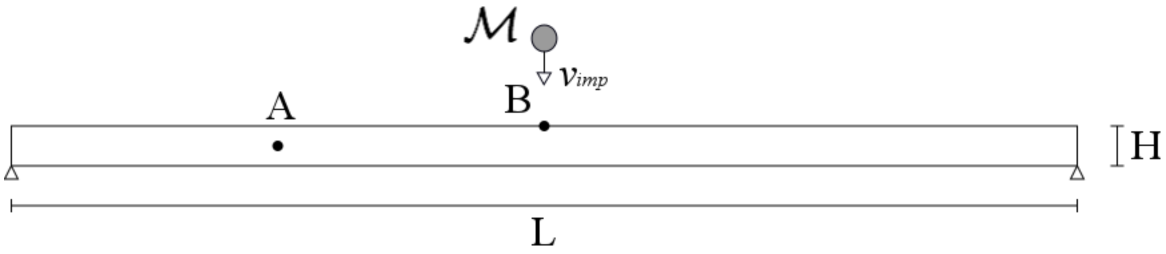
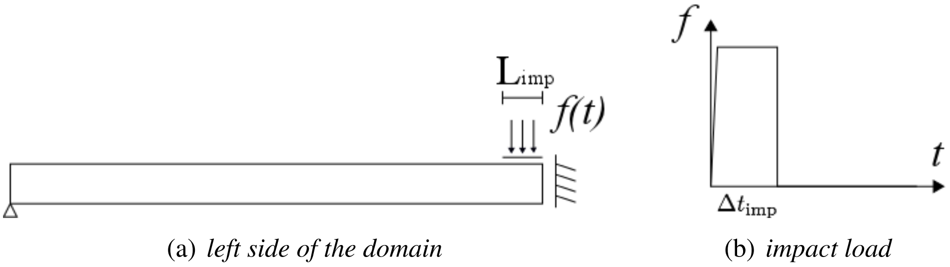


Figure 3.29: Impact test case geometries and monitored locations A and B.

The symmetry of the problem allows to consider the model in Figure 3.30. The horizontal displacements are constrained to zero over the symmetry surface. The impact of the body is modeled as an equivalent impulse load $f(t)$ applied to the surface L_{imp} for the short time interval Δt_{imp} so that:

$$2fL_{\text{imp}}\Delta t_{\text{imp}} = \mathcal{M}v_{\text{imp}} \tag{3.66}$$

assuming that the impacting body does not rebound and its final velocity is approximatively null.



(a) *left side of the domain* (b) *impact load*

Figure 3.30: Impact problem considering the symmetry and simplifying the impact.

Problem data are:

- for the panel:
  - density $\rho = 7800\ kg/m^3$;
  - Young's modulus $E = 210 \cdot 10^9\ Pa$;
  - Poisson's ratio $\nu = 0.3$;
  - length $L = 0.1\ m$;
  - height $H = 0.004\ m$;
- for the impact load:

  - impact time interval $\Delta t_{\text{imp}} = 1 \cdot 10^{-5}$ s starting since t_{in};
  - impact surface of length $2L_{\text{imp}} = 0.01$ m;
  - maximum value $\overline{f} = 10^8$ Pa.
- initial time $t_{\text{in}} = 0$ s;
- final time $t_{\text{end}} = 0.001$ s;

The Domain Decomposition method is performed as reference analysis with the *GCbis* algorithm, decomposing the entire domain in two subdomains as illustrated in Figure 3.31. The first subdomain Ω_1^h with a coarse mesh composed of 8 quadrilateral elements, the second one Ω_2^h with a fine mesh of 40 quadrilateral elements. A finer discretization near the symmetry surface should attenuate the locking effect and catch the transient behavior of the impact load.

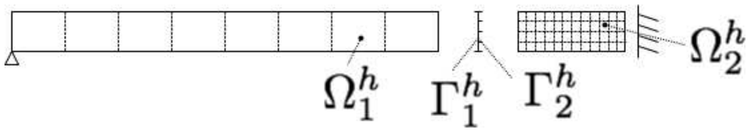


Figure 3.31: Subdomain meshes for the Domain Decomposition.

The Substitution method is performed with the discretizations shown in Figure 3.32. The global analysis is applied to the overall domain with a mesh Ω_g^h of 10 quadrilateral elements, whereas the local one is applied only near the symmetry surface with a mesh Ω_ℓ^h equivalent to the fine subdomain Ω_2^h. The continuity of velocities and the multi-scale equilibrium are applied over the three couples of global and local interfaces Γ_g^h and Γ_ℓ^h, respectively, remarked in Figure 3.32.

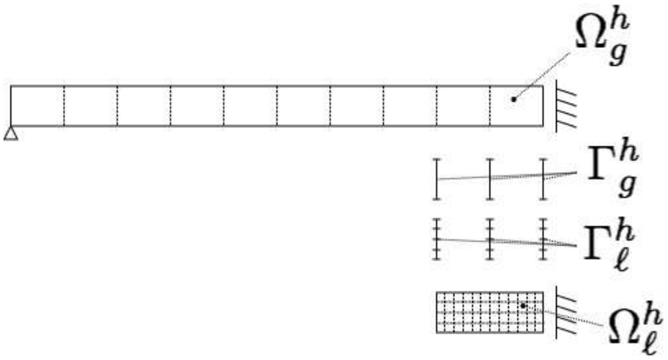


Figure 3.32: Global and local meshes for the Substitution.

According to the mesh sizes, two different time steps are chosen, so that:

- $\Delta t_g = 1 \cdot 10^{-7}$ s for the global analysis and the coarse subdomain,
- $\Delta t_\ell = \Delta t_g / M = 0.25 \cdot 10^{-7}$ s for the local analysis and the fine subdomain,

with $M = h_{g/\ell} = 4$. A convergence tolerance $\overline{e} = 1\%$ is used.

The numerical results of the Substitution method in comparison with the Domain Decomposition method are shown in Figures 3.33 and 3.34 for locations A and B, respectively. Both displacement and velocity curves match at both the locations.

Figure 3.35 shows the convergence rate of the iterative procedure at three representative times of the analysis. The number of iterations required to reach the pre-fixed tolerance $\overline{e}$ is relatively high, even close to 40 iterations. The convergence slope varies during the iterations: from a fast rate for the first 4-5 iterations, it goes down to a slow one for the remaining iterations before that $e < \overline{e}$.

Indeed, the convergence measure e takes into account all the three global-local interfaces in which the solution depends on two different sets of time steps: the correction over the interface at the left between *global complementary* Ω^h_{gc} and *local* Ω^h_ℓ meshes depends on a combination of both the global and local time steps Δt_g and Δt_ℓ, whereas over the two interfaces inside the local mesh Ω^h_ℓ it only depends on the local time step Δt_ℓ. The interfaces are so equally subjected to the kinematic constraint but require different degrees of correction, as the acceleration frequencies inside the local domain and over the global-local interface between the *local* mesh Ω^h_ℓ and the *global complementary* mesh Ω^h_{gc} are different. In particular, the difference between global and local solutions is more remarkable inside the local domain, where local solution frequencies are sensibly higher; on the contrary, the acceleration frequencies over the interface between the *local* Ω^h_ℓ and *global complementary* Ω^h_{gc} meshes are slower because of the contribution of both the global and local time steps Δt_g and Δt_ℓ.

5.1 Acceleration of the iterative scheme: direct substitution

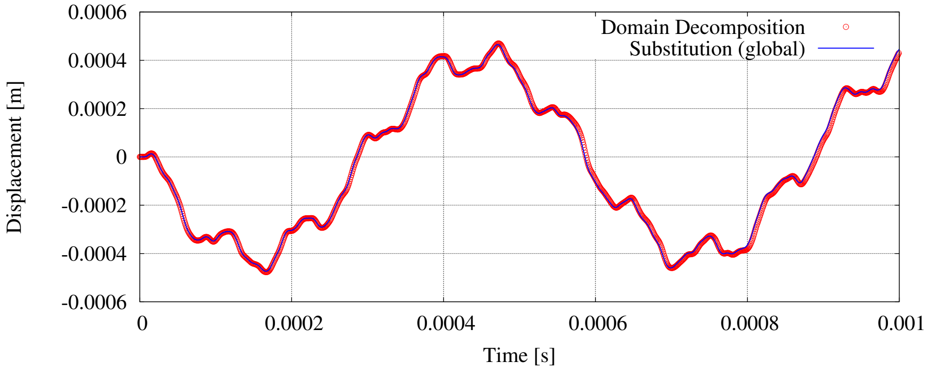
The convergence rates in Figure 3.35 show that the correction in the middle of the local analysis requires more iterations than the correction over the interface between the *global complementary* mesh Ω^h_{gc} and the *local* mesh Ω^h_ℓ. Nevertheless, the interest into obtaining an accurate correction inside the *global substitution* mesh Ω^h_{gs} is minor, because the accurate informations are already contained in the local analysis.

For that reason, a *direct substitution* is applied inside the *global substitution* mesh Ω^h_{gs} in order to accelerate the convergence. The *direct substitution* consists in replacing the global solution in terms of velocities by the mapped local one, so that:

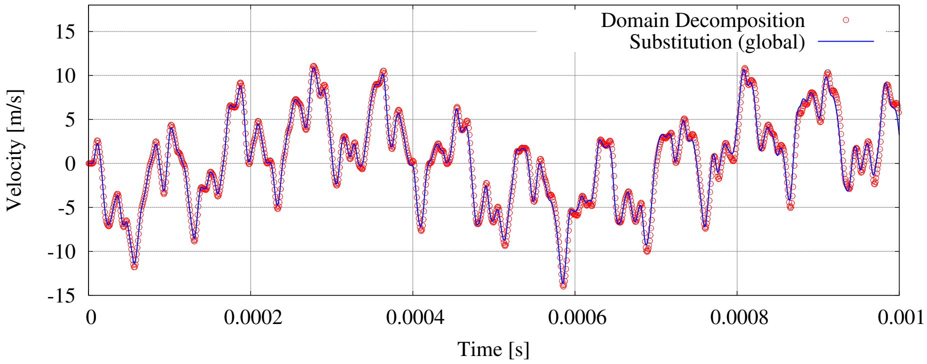
$$\begin{aligned} {}^{n+1}\mathbf{V}_{gs,\mathrm{in}} &= {}^{n+1}\mathbf{V}_{\ell,\mathrm{in}} \\ {}^{n+1}\mathbf{A}_{gs,\mathrm{in}} &= \left({}^{n+1}\mathbf{V}_{\ell,\mathrm{in}} - {}^{n}\mathbf{V}_{gs,\mathrm{in}}\right)\frac{2}{\Delta t_g} - {}^{n}\mathbf{A}_{gs,\mathrm{in}} \end{aligned} \tag{3.67}$$

where the subscripts "gs,in" and "ℓ,in" denote the interior nodes of the meshes Ω^h_{gs} and Ω^h_ℓ, i.e. $\Omega^h_{gs,\mathrm{in}} = \Omega^h_{gs} \setminus \Gamma^h_g$ and $\Omega^h_{\ell,\mathrm{in}} = \Omega^h_\ell \setminus \Gamma^h_\ell$.

So, the *direct substitution* is applied to all the nodes inside the *global substitution* mesh Ω^h_{gs} that do not belong to the *global complementary* mesh Ω^h_{gc}. At those nodes,

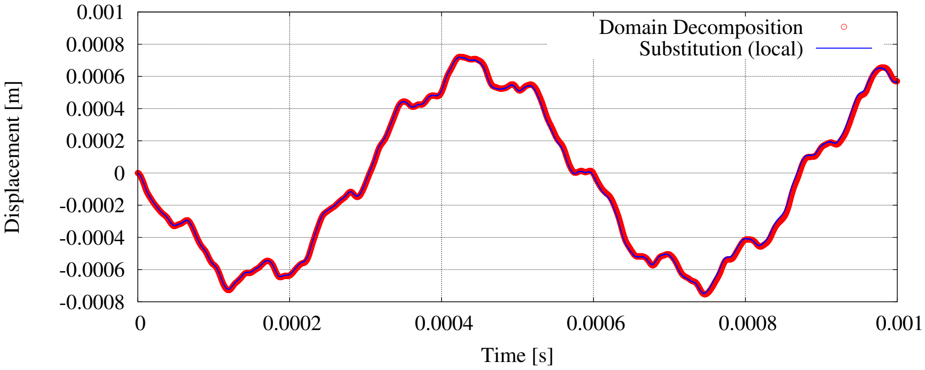


(a) vertical displacement

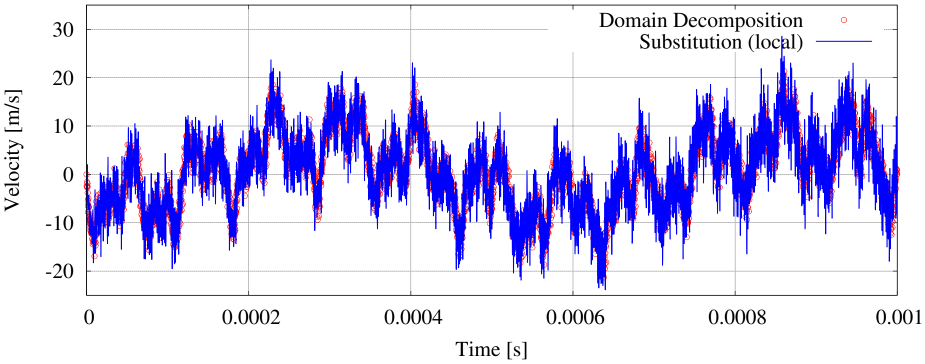


(b) vertical velocity

Figure 3.33: Comparison at the location A between Substitution and Domain Decomposition.



(a) vertical displacement



(b) vertical velocity

Figure 3.34: Comparison at the location B between Substitution and Domain Decomposition.

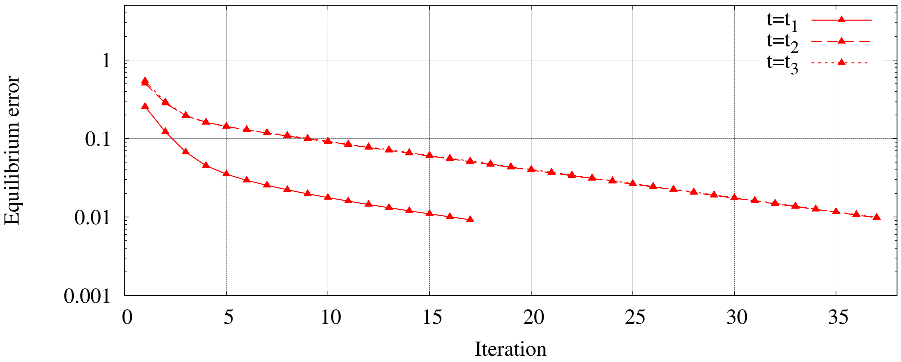


Figure 3.35: Convergence rate with $\bar{e} = 1\%$ at three time instants: $t_i = \frac{1+i}{6} t_{\text{end}}$ with $i = 1,2,3$ and $t_{\text{end}} = 0.001$ s.

the iterative global correction would require more iterations without major benefits to the coupling accuracy.

Figure 3.36 illustrates the new substitution strategy: the two-way *iterative substitution* procedure is applied only to the red nodes over the local interface Γ_ℓ^h between the *local* Ω_ℓ^h and the *global complementary* Ω_{gc}^h meshes, whereas the one-way *direct substitution* is applied to the blue nodes inside the *global substitution* mesh Ω_{gs}^h.

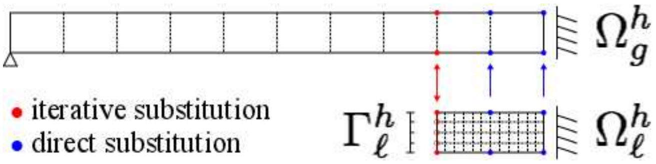


Figure 3.36: Procedure with direct substitution.

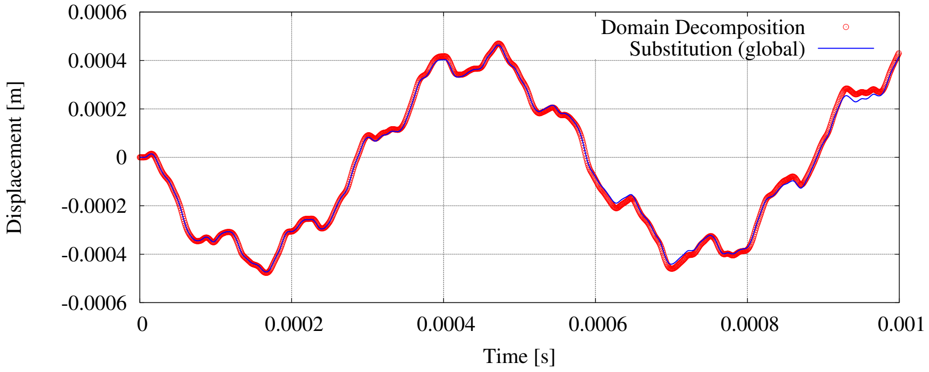
Figures 3.37 and 3.38 show the acceptable matching between the Substitution and Domain Decomposition, also at the location B where the *direct substitution* is applied.

The gain in terms of number of iterations is shown in Figure 3.39. The approach with the iterative scheme over all the three interfaces required a number of iterations varying between 10 and 40, whereas the approach utilizing the *direct substitution* requires between 5 and 9 iterations.

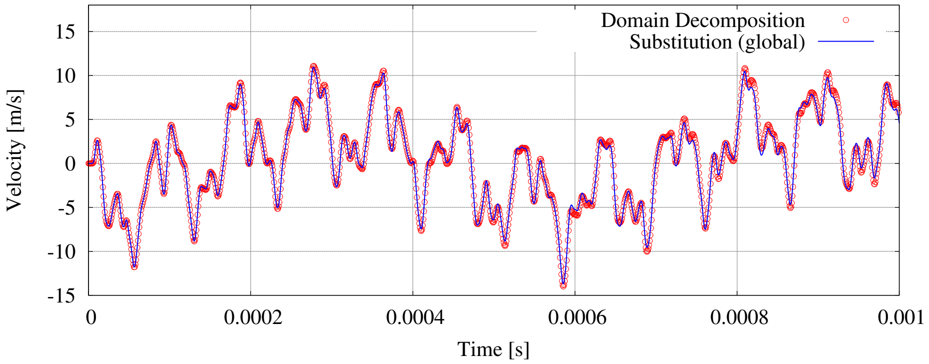
Nonetheless, such number of iterations remains too high for the industrial application. Further studies are so dedicated to the improvement of the iterative scheme.

6 Conclusions

To conclude, in this Chapter, the Substitution method has been formulated and studied in comparison with the Domain Decomposition method. The Substitution could be advantageous because it avoids cutting the topology of the problem. For this reason, it gains the

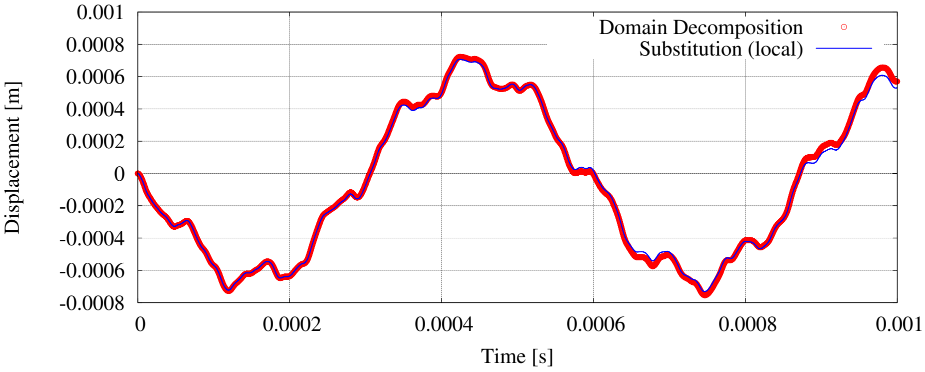


(a) vertical displacement

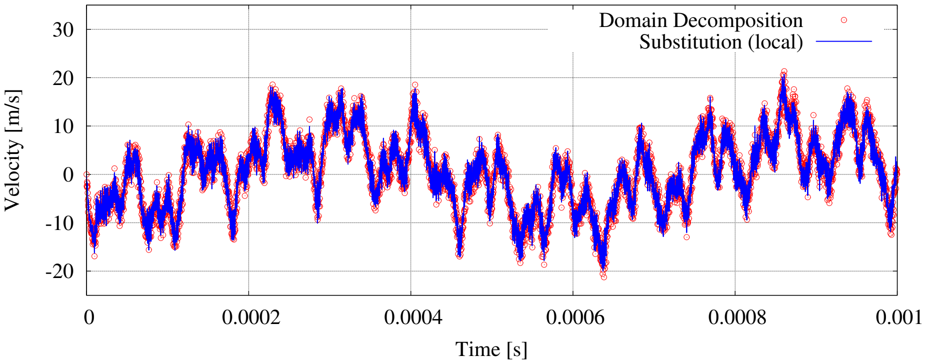


(b) vertical velocity

Figure 3.37: Comparison at the location A between Substitution and Domain Decomposition.



(a) vertical displacement



(b) vertical velocity

Figure 3.38: Comparison at the location B between Substitution and Domain Decomposition.

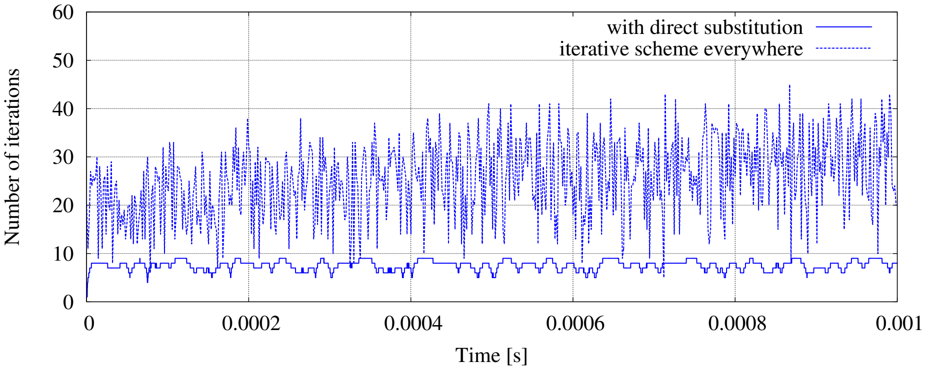


Figure 3.39: Gain with the direct substitution approach in terms of number of iterations during the impact simulation with $\overline{e} = 1\%$.

characteristic of weakly-intrusive for such dynamic cases with evolutive phenomena that require a dynamic adaptivity of the local model, keeping the global analysis unchanged and driving the activation of local patches where necessary.

A cubic Hermitian interpolation has been chosen as time down-scaling operator from global to local. Although it does not ensure the displacement continuity, it allows the continuity of velocities and accelerations at the global level.

The stability of the coupling has been studied with the energy method as for the multi-time-step FETI approaches. An energy-preserving methodology based on the same developments in Domain Decomposition has been proposed. Nevertheless, the extension of such methodology to nonlinearities or to adaptive time-stepping is computationally expensive.

The original algorithm has been preferred and the *direct substitution* is introduced in order to reduce the number of iterations in case of local domains overlapping more global elements. However, the number of iterations remains too elevated to obtain acceptable performances.

The algorithm taken as starting point for further enhancements is summarized in Algorithm 11. A rearrangement of the steps is employed here in order to avoid the residual computation of the accelerations ${}^{n+1}\mathbf{A}_{g,\Gamma}^{\mathrm{resid}}$: the global correction is enforced before the local analysis so that both the global and local solutions are related to iteration i at the end of each iterative step.

```
while $t_g < t_{end}$ do
    global pre-computation: $\mathbf{M}_g {}^{n+1}\mathbf{A}_g^{\text{prec}} = {}^{n+1}\mathbf{F}_g^{\text{ext}} - {}^{n+1}\mathbf{F}_g^{\text{int}}$;
    if Substitution is activated then
        initialize $\left({}^{n+1}\mathbf{A}_{g,\Gamma}\right)^0 \leftarrow {}^{n+1}\mathbf{A}_{g,\Gamma}^{\text{prec}}$;
        while $t_\ell < t_g + \Delta t_g$ do
            local pre-computation: $\mathbf{M}_\ell \left({}^{m+1}\mathbf{A}_\ell\right)^0 = {}^{m+1}\mathbf{F}_\ell^{\text{ext}} - \left({}^{m+1}\mathbf{F}_\ell^{\text{int}}\right)^0$;
            with: ${}^{m+1}\mathbf{V}_{\ell,\Gamma} \leftarrow \Pi_h^1 \Pi_t^3 \left({}^{n}\mathbf{V}_{g,\Gamma}, {}^{n}\mathbf{A}_{g,\Gamma}, \left({}^{n+1}\mathbf{V}_{g,\Gamma}\right)^0, \left({}^{n+1}\mathbf{A}_{g,\Gamma}\right)^0\right)$;
            update $t_\ell \leftarrow t_\ell + \Delta t_\ell$;
        end
        while $e > \bar{e}$ do
            global correction: $\mathbf{M}_{g,\Gamma} \left({}^{n+1}\mathbf{A}_{g,\Gamma}^{\text{corr}}\right)^i = \left({}^{n+1}\mathbf{P}\right)^i$;
            update $\left({}^{n+1}\mathbf{A}_{g,\Gamma}\right)^i \leftarrow {}^{n+1}\mathbf{A}_{g,\Gamma}^{\text{prec}} + \left({}^{n+1}\mathbf{A}_{g,\Gamma}^{\text{corr}}\right)^i$;
            back in time $t_\ell \leftarrow t_g$;
            while $t_\ell < t_g + \Delta t_g$ do
                local computation: $\mathbf{M}_\ell \left({}^{m+1}\mathbf{A}_\ell\right)^i = {}^{m+1}\mathbf{F}_\ell^{\text{ext}} - \left({}^{m+1}\mathbf{F}_\ell^{\text{int}}\right)^i$;
                with:
                ${}^{m+1}\mathbf{V}_{\ell,\Gamma} \leftarrow \Pi_h^1 \Pi_t^3 \left({}^{n}\mathbf{V}_{g,\Gamma}, {}^{n}\mathbf{A}_{g,\Gamma}, \left({}^{n+1}\mathbf{V}_{g,\Gamma}\right)^i, \left({}^{n+1}\mathbf{A}_{g,\Gamma}\right)^i\right)$;
                update $t_\ell \leftarrow t_\ell + \Delta t_\ell$;
            end
            update $i \leftarrow i + 1$;
            compute $e$;
        end
        update ${}^{n+1}\mathbf{V}_{gs,\text{in}} \leftarrow {}^{n+1}\mathbf{V}_{\ell,\text{in}}$;
        update ${}^{n+1}\mathbf{A}_{gs,\text{in}} \leftarrow \left({}^{n+1}\mathbf{V}_{\ell,\text{in}} - {}^{n}\mathbf{V}_{gs,\text{in}}\right) \frac{2}{\Delta t_g} - {}^{n}\mathbf{A}_{gs,\text{in}}$;
    end
    update $t_g \leftarrow t_g + \Delta t_g$;
end
```

Algorithm 11: Iterative substitution algorithm in a dynamic analysis avoiding the residual addition and involving the direct substitution.

Chapter 4

Enhancement of the iterative scheme

In this Chapter as in [Bettinotti et al., 2014b], the enhancement of the iterative procedure is described. Based on compatibility properties, a new algorithm procedure is defined. New test cases involving defects and nonlinearities are presented and an adaptive case is introduced. The reduction of the required number of iterations is always evidenced.

Contents

1 Introduction

The Substitution method proposed in Chapter 3 is based on an iterative technique: the explicit character of the method allows to perform the global computation only once per global time step, while a repeated solution is required for the small local problems only. Nevertheless, a desirable goal is to reach convergence with a reduced number of iterations.

To this purpose, we propose here a new iterative algorithm based on a property that was not noticed at first: the coupling of the cubic Hermitian interpolation in time, enforcing the continuity of velocities, with the application of the Central Difference scheme to the global and local analyses leads to the equality of accelerations. Such equality allows one to define an updated mass operator for the interface coupling, significantly enhancing the convergence rate. The efficiency and robustness of the procedure is demonstrated through several examples of growing complexity.

2 Time down-scaling operator property

Out of the study of Section 3.4 in Chapter 3, the cubic Hermitian interpolation emerged as a way of imposing the local velocities obtaining the continuity of both velocities and accelerations at the global time scale. Recalling Equation (3.33):

$$\begin{aligned}\dot{\mathbf{U}}_{\ell,\Gamma}\big|_{[t^n,t^{n+1}]} = {} & \Pi_h^1\, {}^{n}\mathbf{V}_{g,\Gamma}\left(1-3s^2+2s^3\right) + \Pi_h^1\, {}^{n}\mathbf{A}_{g,\Gamma}\Delta t_g\left(s-2s^2+s^3\right) + \\ & + \Pi_h^1\, {}^{n+1}\mathbf{V}_{g,\Gamma}\left(3s^2-2s^3\right) + \Pi_h^1\, {}^{n+1}\mathbf{A}_{g,\Gamma}\Delta t_g\left(-s^2+s^3\right).\end{aligned} \tag{4.1}$$

Nevertheless, when used together with the Central Difference scheme at both the scales, the cubic Hermitian interpolation of velocities exhibits an additional crucial property, which has not been noticed up to this point.

According to the Central Difference scheme, the global velocities at the end of the time step are expressed in terms of the previous velocities and accelerations as:

$${}^{n+1}\mathbf{V}_{g,\Gamma} = {}^{n}\mathbf{V}_{g,\Gamma} + \left({}^{n}\mathbf{A}_{g,\Gamma} + {}^{n+1}\mathbf{A}_{g,\Gamma}\right)\frac{\Delta t_g}{2}. \tag{4.2}$$

Equation (4.2) shows that the current global velocities ${}^{n+1}\mathbf{V}_{g,\Gamma}$ are not an independent parameter in Equation (4.1) and can be there replaced in function of the other three parameters ${}^{n}\mathbf{V}_{g,\Gamma}$, ${}^{n}\mathbf{A}_{g,\Gamma}$ and ${}^{n+1}\mathbf{A}_{g,\Gamma}$. Equation (4.1) so becomes:

$$\dot{\mathbf{U}}_{\ell,\Gamma}\big|_{[t^n,t^{n+1}]} = \Pi_h^1\, {}^{n}\mathbf{V}_{g,\Gamma} + \Pi_h^1\, {}^{n}\mathbf{A}_{g,\Gamma}\Delta t_g s + \Pi_h^1\left({}^{n+1}\mathbf{A}_{g,\Gamma} - {}^{n}\mathbf{A}_{g,\Gamma}\right)\Delta t_g s^2/2 \tag{4.3}$$

which reveals that a quadratic shape is actually enforced to velocities at the local scale, that in the local time stepping reads:

$${}^{m}\mathbf{V}_{\ell,\Gamma} = \Pi_h^1\, {}^{n}\mathbf{V}_{g,\Gamma} + \Pi_h^1\, {}^{n}\mathbf{A}_{g,\Gamma}\Delta t_g\frac{m}{M} + \Pi_h^1\left({}^{n+1}\mathbf{A}_{g,\Gamma} - {}^{n}\mathbf{A}_{g,\Gamma}\right)\frac{\Delta t_g}{2}\frac{m^2}{M^2}. \tag{4.4}$$

Deriving the imposed local velocities in function of time, such that:

$$\ddot{\mathbf{U}}_{\ell,\Gamma}\big|_{[t^n,t^{n+1}]} = \frac{1}{\Delta t_g}\frac{\partial \dot{\mathbf{U}}_{\ell,\Gamma}}{\partial s}\bigg|_{[t^n,t^{n+1}]} \tag{4.5}$$

Equation (4.3) becomes:

$$\ddot{\mathbf{U}}_{\ell,\Gamma}\big|_{[t^n,t^{n+1}]} = \Pi_h^1\, {}^{n}\mathbf{A}_{g,\Gamma} + \Pi_h^1 \left({}^{n+1}\mathbf{A}_{g,\Gamma} - {}^{n}\mathbf{A}_{g,\Gamma}\right) s \tag{4.6}$$

which shows a linear interpolation between ${}^{n}\mathbf{A}_{g,\Gamma}$ and ${}^{n+1}\mathbf{A}_{g,\Gamma}$, while in the local time steps reads:

$${}^{m}\mathbf{A}_{\ell,\Gamma} = \Pi_h^1\, {}^{n}\mathbf{A}_{g,\Gamma}\left(1 - \frac{m}{M}\right) + \Pi_h^1\, {}^{n+1}\mathbf{A}_{g,\Gamma}\left(\frac{m}{M}\right). \tag{4.7}$$

This implies that the local and global accelerations over the interface coincide at each global time step, i.e.:

$${}^{M}\mathbf{A}_{\ell,\Gamma} = \Pi_h^1\, {}^{n+1}\mathbf{A}_{g,\Gamma}. \tag{4.8}$$

Note that this is not the case in Domain Decomposition where continuity is enforced by assuming the equality of velocities which, in general, does not imply the acceleration's equality.

Nevertheless, Equations (4.4) and (4.7) must also satisfy the local Central Difference scheme, i.e.:

$${}^{m+1}\mathbf{V}_{\ell,\Gamma} - {}^{m}\mathbf{V}_{\ell,\Gamma} = \left({}^{m}\mathbf{A}_{\ell,\Gamma} + {}^{m+1}\mathbf{A}_{\ell,\Gamma}\right)\frac{\Delta t_\ell}{2} \quad \forall m \in \{0,\dots,M-1\}. \tag{4.9}$$

Taking into account the definition of local velocities in Equation (4.4), the velocities difference in the right-hand side of Equation (4.9) reads:

$${}^{m+1}\mathbf{V}_{\ell,\Gamma} - {}^{m}\mathbf{V}_{\ell,\Gamma} = \Pi_h^1\, {}^{n}\mathbf{A}_{g,\Gamma}\frac{\Delta t_g}{M} + \Pi_h^1 \left({}^{n+1}\mathbf{A}_{g,\Gamma} - {}^{n}\mathbf{A}_{g,\Gamma}\right)\frac{\Delta t_g}{2}\frac{(2m+1)}{M^2}. \tag{4.10}$$

On the other hand, taking into account the definition of accelerations in Equation (4.7), the accelerations sum at the left-hand side of Equation (4.9) reads:

$${}^{m}\mathbf{A}_{\ell,\Gamma} + {}^{m+1}\mathbf{A}_{\ell,\Gamma} = \Pi_h^1\, {}^{n}\mathbf{A}_{g,\Gamma}\left(2 - \frac{2m+1}{M}\right) + \Pi_h^1\, {}^{n+1}\mathbf{A}_{g,\Gamma}\frac{2m+1}{M}. \tag{4.11}$$

Substituting the definitions in Equations (4.10) and (4.11) in Equation (4.9), it follows that:

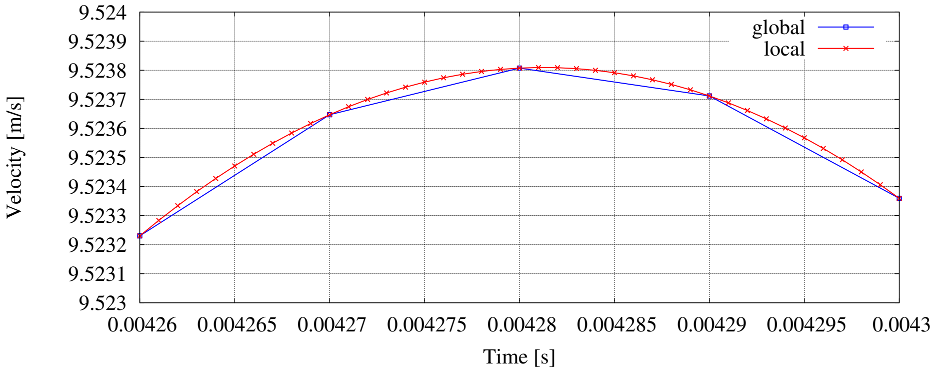
$$\begin{aligned} &\Pi_h^1\, {}^{n}\mathbf{A}_{g,\Gamma}\frac{\Delta t_g}{M} + \Pi_h^1 \left({}^{n+1}\mathbf{A}_{g,\Gamma} - {}^{n}\mathbf{A}_{g,\Gamma}\right)\frac{\Delta t_g}{2}\frac{(2m+1)}{M^2} = \\ &\quad = \left(\Pi_h^1\, {}^{n}\mathbf{A}_{g,\Gamma}\left(2 - \frac{2m+1}{M}\right) + \Pi_h^1\, {}^{n+1}\mathbf{A}_{g,\Gamma}\frac{2m+1}{M}\right)\frac{\Delta t_\ell}{2} \end{aligned} \tag{4.12}$$

that, considering $\Delta t_g = M\Delta t_\ell$, is rearranged as:

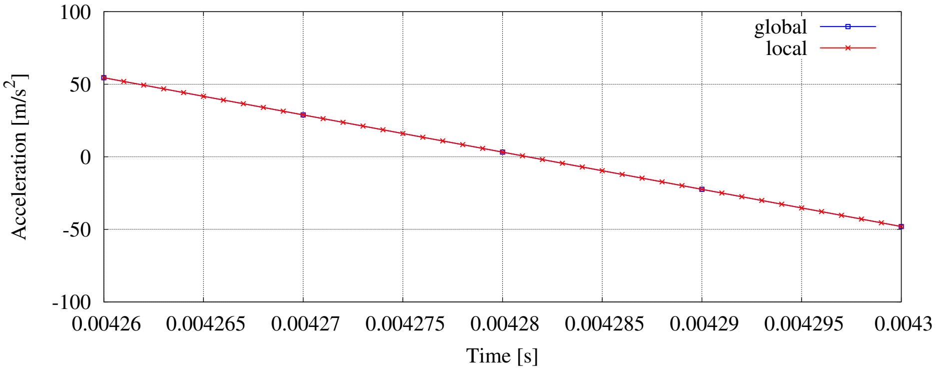
$$\begin{aligned} 2\Pi_h^1\, {}^n\mathbf{A}_{g,\Gamma} + \Pi_h^1 \left({}^{n+1}\mathbf{A}_{g,\Gamma} - {}^n\mathbf{A}_{g,\Gamma}\right) \frac{2m+1}{M} &= \\ = \Pi_h^1\, {}^n\mathbf{A}_{g,\Gamma} \left(2 - \frac{2m+1}{M}\right) + \Pi_h^1\, {}^{n+1}\mathbf{A}_{g,\Gamma} \frac{2m+1}{M} & \end{aligned} \tag{4.13}$$

which, being verified $\forall m \in \{0, \ldots, M-1\}$, asserts the compatibility of the time down-scaling operator and the local Central Difference scheme.

The spring application described in Figure 3.8 in Section 3.4 of Chapter 3 is so considered in order to verify the equality between accelerations. Figure 4.1 recalls the continuity of velocities at the global level and shows the continuity of accelerations even at the local level.



(a) velocities



(b) accelerations

Figure 4.1: Kinematic constraint with a cubic Hermitian interpolation in time and applying the Central Difference scheme for global and local solutions.

3 Reformulation of the interface multi-scale compatibility condition

When the local region is replaced by the global substitution region as in Figure 3.6 in Section 3.1 of Chapter 3, the *global interface momentum balance* is integrated in time as in Equation (3.11) and the *correction forces* ${}^{n+1}\mathbf{P}$ are determined as in Equation (3.12). Recalling the equations, the *global interface momentum balance* reads:

$$\mathbf{M}_{g,\Gamma}{}^{n+1}\mathbf{A}_{g,\Gamma} = {}^{n+1}\mathbf{F}^{\text{ext}}_{g,\Gamma} - {}^{n+1}\mathbf{F}^{\text{int}}_{g,\Gamma} + {}^{n+1}\mathbf{P} \tag{4.14}$$

where the *correction forces* read:

$${}^{n+1}\mathbf{P} = \Pi_h^{1T}\,{}^{n+1}\mathbf{R}_{\ell,\Gamma} - {}^{n+1}\mathbf{R}_{gs,\Gamma}. \tag{4.15}$$

Reporting once again the splitting definitions in Equation (3.9), Equation (4.14) is also split in global substitution and complementary terms, as:

$$\left(\mathbf{M}_{gs,\Gamma} + \mathbf{M}_{gc,\Gamma}\right){}^{n+1}\mathbf{A}_{g,\Gamma} = {}^{n+1}\mathbf{F}^{\text{ext}}_{gs,\Gamma} + {}^{n+1}\mathbf{F}^{\text{ext}}_{gc,\Gamma} + {}^{n+1}\mathbf{P} + {}^{n+1}\mathbf{F}^{\text{ext}}_{\Gamma,\Gamma} - {}^{n+1}\mathbf{F}^{\text{int}}_{gs,\Gamma} - {}^{n+1}\mathbf{F}^{\text{int}}_{gc,\Gamma}. \tag{4.16}$$

On the other hand, proposing another time the definitions of the reactions forces ${}^{n+1}\mathbf{R}_{\ell,\Gamma}$ and ${}^{n+1}\mathbf{R}_{gs,\Gamma}$ in Equations (3.3) and (3.5), respectively, Equation (4.15) can be rewritten as:

$${}^{n+1}\mathbf{P} = \Pi_h^{1T}\left({}^{n+1}\mathbf{F}^{\text{ext}}_{\ell,\Gamma} - \mathbf{M}_{\ell,\Gamma}{}^{n+1}\mathbf{A}_{\ell,\Gamma} - {}^{n+1}\mathbf{F}^{\text{int}}_{\ell,\Gamma}\right) - {}^{n+1}\mathbf{F}^{\text{ext}}_{gs,\Gamma} + \mathbf{M}_{gs,\Gamma}{}^{n+1}\mathbf{A}_{g,\Gamma} + {}^{n+1}\mathbf{F}^{\text{int}}_{gs,\Gamma} \tag{4.17}$$

and applying the acceleration compatibility property of Equation (4.8) of Section 2, one obtains the new definition:

$${}^{n+1}\mathbf{P} = \left(\mathbf{M}_{gs,\Gamma} - \Pi_h^{1T}\mathbf{M}_{\ell,\Gamma}\Pi_h^1\right){}^{n+1}\mathbf{A}_{g,\Gamma} - {}^{n+1}\mathbf{F}^{\text{ext}}_{gs,\Gamma} + {}^{n+1}\mathbf{F}^{\text{int}}_{gs,\Gamma} + \Pi_h^{1T}\left({}^{n+1}\mathbf{F}^{\text{ext}}_{\ell,\Gamma} - {}^{n+1}\mathbf{F}^{\text{int}}_{\ell,\Gamma}\right) \tag{4.18}$$

where $\Pi_h^{1T}\mathbf{M}_{\ell,\Gamma}\Pi_h^1$ is the up-scaled term of the local interface mass to the global interface.

Substituting the *correction forces* definition of Equation (4.18) into the *global interface momentum balance* of Equation (4.16), one obtains:

$$\begin{aligned}\left(\Pi_h^{1T}\mathbf{M}_{\ell,\Gamma}\Pi_h^1 + \mathbf{M}_{gc,\Gamma}\right){}^{n+1}\mathbf{A}_{g,\Gamma} = {}& \Pi_h^{1T}\left({}^{n+1}\mathbf{F}^{\text{ext}}_{\ell,\Gamma} - {}^{n+1}\mathbf{F}^{\text{int}}_{\ell,\Gamma}\right) + \\ & + {}^{n+1}\mathbf{F}^{\text{ext}}_{gc,\Gamma} - {}^{n+1}\mathbf{F}^{\text{int}}_{gc,\Gamma} + {}^{n+1}\mathbf{F}^{\text{ext}}_{\Gamma,\Gamma}\end{aligned} \tag{4.19}$$

that denotes the momentum balance over the interface of the *reference problem* considering that local internal forces constitute a result of the local solution obtained in the fine time scale.

4 Improved iterative scheme

Some of the terms in Equation (4.19) can be computed only once per global time step, using informations coming from the global problem or from the solution of the previous time step, whereas others have to be computed iteratively. Terms of the first type are used to define *pre-computation* values of the acceleration ${}^{n+1}\mathbf{A}^{\text{prec}}_{g,\Gamma}$, so that:

$$\left(\Pi_h^{1T}\mathbf{M}_{\ell,\Gamma}\Pi_h^1+\mathbf{M}_{gc,\Gamma}\right){}^{n+1}\mathbf{A}^{\text{prec}}_{g,\Gamma}=\Pi_h^{1T}\,{}^{n+1}\mathbf{F}^{\text{ext}}_{\ell,\Gamma}+{}^{n+1}\mathbf{F}^{\text{ext}}_{gc,\Gamma}-{}^{n+1}\mathbf{F}^{\text{int}}_{gc,\Gamma}+{}^{n+1}\mathbf{F}^{\text{ext}}_{\Gamma,\Gamma} \quad (4.20)$$

while terms of the second type provide iterative *correction* values of acceleration $\left({}^{n+1}\mathbf{A}^{\text{corr}}_{g,\Gamma}\right)^i$, that are:

$$\left(\Pi_h^{1T}\mathbf{M}_{\ell,\Gamma}\Pi_h^1+\mathbf{M}_{gc,\Gamma}\right)\left({}^{n+1}\mathbf{A}^{\text{corr}}_{g,\Gamma}\right)^i=-\Pi_h^{1T}\left({}^{n+1}\mathbf{F}^{\text{int}}_{\ell,\Gamma}\right)^i. \quad (4.21)$$

In Equation (4.20), ${}^{n+1}\mathbf{F}^{\text{int}}_{gc,\Gamma}$ is function of quantities of the previous step, since in the Central Difference scheme the nodal displacements are an outcome of the explicit integration and are computed directly from the known accelerations. In Equation (4.21), $\left({}^{n+1}\mathbf{F}^{\text{int}}_{\ell,\Gamma}\right)^i$ are the updated internal forces of the local problem over the interface Γ^h_ℓ, computed on the basis of the current estimate of acceleration ${}^{n+1}\mathbf{A}^{\text{prec}}_{g,\Gamma}+\left({}^{n+1}\mathbf{A}^{\text{corr}}_{g,\Gamma}\right)^i$. At the end of the pre-computation phase, the local internal forces are initialized assigning over the interface the pre-computed value ${}^{n+1}\mathbf{A}^{\text{prec}}_{g,\Gamma}$ only.

Both Equations (4.20) and (4.21) require the resolution of a linear system with a non-diagonal coefficients matrix. The terms out of diagonal of such matrix come from the up-scaling of the local mass matrix to the global interface Γ^h_g with $\Pi_h^{1T}\mathbf{M}_{\ell,\Gamma}\Pi_h^1$. Nevertheless, the number of degrees of freedom is associated to the number of nodes over the global interface, in contrast with the down-scaling operation to the local interface Γ^h_ℓ in the condensation matrix of the Domain Decomposition.

As in the fixed-point iterative algorithm described in Section 3.3 of Chapter 3, the iterative procedure is stopped when a pre-fixed tolerance $\bar{e}$ on the error on interface equilibrium at the global level is met, referring to the convergence measure e defined in Equation (3.22).

Compared to the direct solution of the *reference problem*, the proposed iterative multi-scale approach implies at each time step the solution of the overall global problem, which however has been discretized by means of a coarse mesh and possibly with a simplified geometry and material behavior. It is therefore expected to be a problem significantly less expensive than the original one. On the other hand, the local problem, requires an accurate description of geometry and material behavior and a fine mesh, but concerns a small part of the problem domain, so that its iterative solution can be carried out with little computational effort.

The flow chart of the proposed improved procedure is described in Algorithm 12.

```
while $t_g < t_{end}$ do
    global pre-computation: $\mathbf{M}_g {}^{n+1}\mathbf{A}_g^{\text{prec}} = {}^{n+1}\mathbf{F}_g^{\text{ext}} - {}^{n+1}\mathbf{F}_g^{\text{int}}$;
    if Substitution is activated then
        global re-pre-computation:
        $\left(\mathbf{M}_{gc,\Gamma} + \Pi_h^{1T}\mathbf{M}_{\ell,\Gamma}\Pi_h^1\right) {}^{n+1}\mathbf{A}_{g,\Gamma}^{\text{prec}} = \Pi_h^{1T\,n+1}\mathbf{F}_{\ell,\Gamma}^{\text{ext}} + {}^{n+1}\mathbf{F}_{gc,\Gamma}^{\text{ext}} - {}^{n+1}\mathbf{F}_{gc,\Gamma}^{\text{int}}$;
        while $t_\ell < t_g + \Delta t_g$ do
            local computation (iteration 0):
            $\mathbf{M}_\ell \left({}^{m+1}\mathbf{A}_\ell\right)^0 = {}^{m+1}\mathbf{F}_\ell^{\text{ext}} - {}^{m+1}\mathbf{F}_\ell^{\text{int}}$;
            with: ${}^{m+1}\mathbf{V}_{\ell,\Gamma} \leftarrow \Pi_h^1\Pi_t^3 \left({}^n\mathbf{V}_{g,\Gamma}, {}^n\mathbf{A}_{g,\Gamma}, {}^{n+1}\mathbf{A}_{g,\Gamma}^{\text{prec}}\right)$;
            update $t_\ell \leftarrow t_\ell + \Delta t_\ell$;
        end
        update $i \leftarrow 1$;
        while $e > \bar{e}$ do
            back in time $t_\ell \leftarrow t_g$;
            global correction:
            $\left(\mathbf{M}_{gc,\Gamma} + \Pi_h^{1T}\mathbf{M}_{\ell,\Gamma}\Pi_h^1\right) \left({}^{n+1}\mathbf{A}_{g,\Gamma}\right)^i = -\Pi_h^{1T} \left({}^{n+1}\mathbf{F}_{\ell,\Gamma}^{\text{int}}\right)^i$;
            update $\left({}^{n+1}\mathbf{A}_{g,\Gamma}\right)^i \leftarrow {}^{n+1}\mathbf{A}_{g,\Gamma}^{\text{prec}} + \left({}^{n+1}\mathbf{A}_{g,\Gamma}^{\text{corr}}\right)^i$;
            while $t_\ell < t_g + \Delta t_g$ do
                local computation (iteration $i$):
                $\mathbf{M}_\ell \left({}^{m+1}\mathbf{A}_\ell\right)^i = {}^{m+1}\mathbf{F}_\ell^{\text{ext}} - {}^{m+1}\mathbf{F}_\ell^{\text{int}}$;
                with: ${}^{m+1}\mathbf{V}_{\ell,\Gamma} \leftarrow \Pi_h^1\Pi_t^3 \left({}^n\mathbf{V}_{g,\Gamma}, {}^n\mathbf{A}_{g,\Gamma}, \left({}^{n+1}\mathbf{A}_{g,\Gamma}\right)^i\right)$;
                update $t_\ell \leftarrow t_\ell + \Delta t_\ell$;
            end
            update $i \leftarrow i + 1$;
            compute $e$;
        end
        update ${}^{n+1}\mathbf{V}_{gs,\text{in}} \leftarrow {}^{n+1}\mathbf{V}_{\ell,i}$;
        update ${}^{n+1}\mathbf{A}_{gs,\text{in}} \leftarrow \left({}^{n+1}\mathbf{V}_{\ell,i} - {}^n\mathbf{V}_{gs,\text{in}}\right) \frac{2}{\Delta t_g} - {}^n\mathbf{A}_{gs,\text{in}}$;
    end
    update $t_g \leftarrow t_g + \Delta t_g$;
end
```

Algorithm 12: Substitution algorithm with enhanced iterative scheme.

5 Applications

The performance of the proposed algorithm is illustrated by means of three examples:

1. *local* region consisting of a mesh refinement of the *global substitution* region;

2. *local* region with defects and heterogeneities of various types;

3. model adaptation in order to follow the delamination propagation.

In all cases, the results are compared with the original version of the Substitution method presented in Chapter 3, in order to comparatively assess the improved convergence properties. The example with heterogeneities is used to show the robustness of the proposed method both in terms of quality of results when compared to a Domain Decomposition method and in terms of convergence rate. The last example is a first attempt to apply the method to a more complex case involving nonlinearities and model adaptation.

The tolerance criteria for the convergence measure e computed in Equation (3.22) is set at $\bar{e} = 1\%$. In fact, it was shown in Chapter 3 that with such a tolerance the results obtained in terms of velocity are in agreement with those obtained by a Domain Decomposition approach.

5.1 Local mesh refinement

As in Chapter 3, the elastic global model of the structure is discretized by a coarse mesh of 2 displacement-based 4-node quadrilateral elements. The local model consists of a refined mesh made of smaller quadrilateral elements of the same type (as in Figure 4.2) concerning only a limited region of the structure, close to the fixed boundary, in order to attenuate shear locking effects.

The linear elastic problem data are:

- density $\rho = 7800\ kg/m^3$;

- Young's modulus $E = 210 \cdot 10^9\ Pa$;

- Poisson's ratio $\nu = 0.3$;

- length $L = 30\ m$;

- height $H = 10\ m$;

- uniform shear load $\overline{f} = 3 \cdot 10^6\ N/m^2$ constant in time.

The results in terms of accelerations on time are observed at two representative locations of the domain (Figure 4.2): the location A at the bottom free corner of the structure and the location B at the bottom of the interface between the global and local regions.

The time steps at the two scales are chosen so that:

- $\Delta t_g = 8 \cdot 10^{-4}\ s$ in the global analysis;

- $\Delta t_\ell = \Delta t_g / M = 0.5 \cdot 10^{-4}\ s$ in the local analysis;

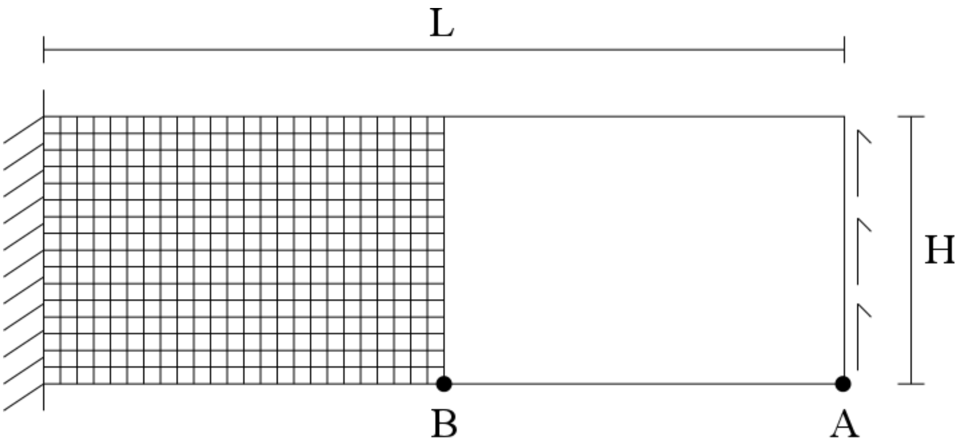
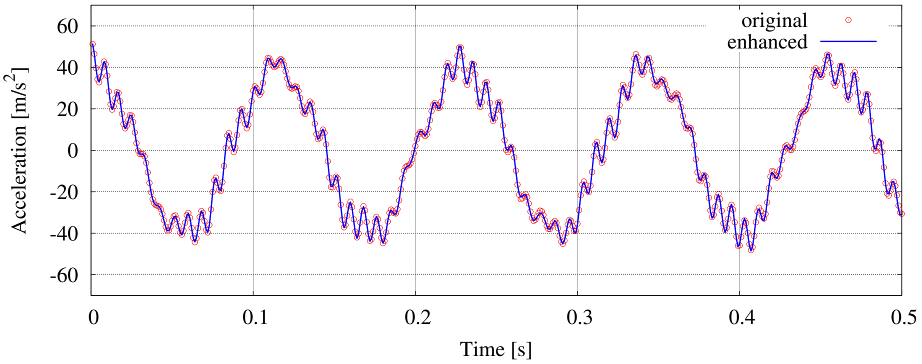
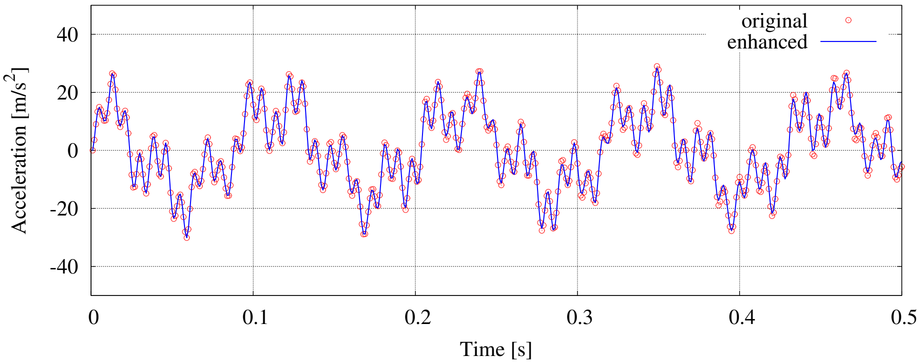


Figure 4.2: Spatial discretization, sizes and monitored locations A and B.



(a) location A



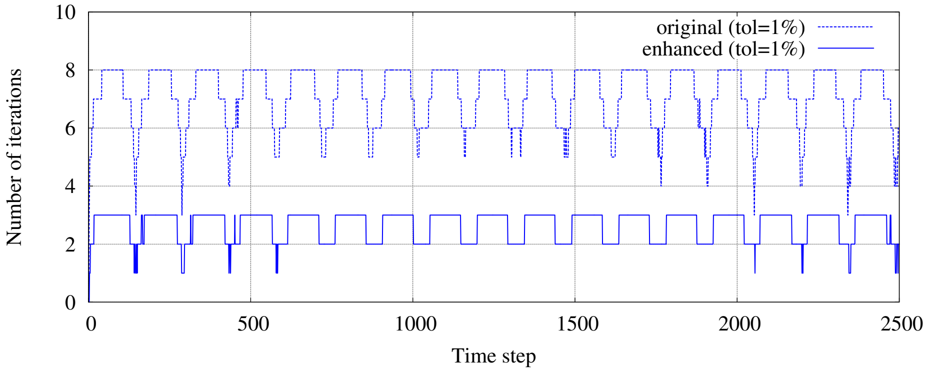
(b) location B

Figure 4.3: Comparison of the vertical accelerations between the original and enhanced Substitution methods.

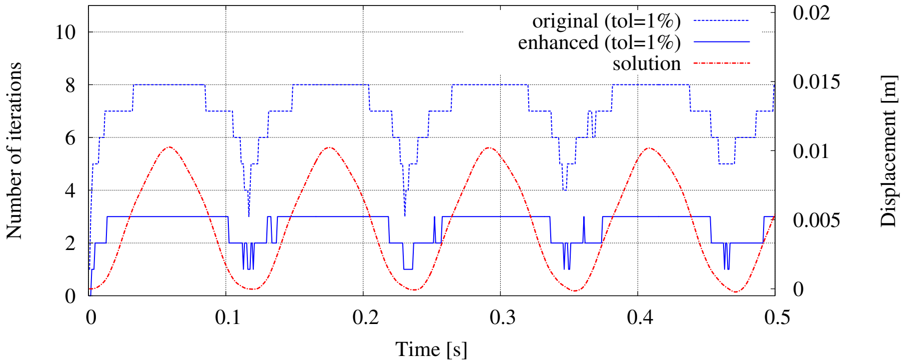
with $M = h_{g/\ell} = 16$.

Figure 4.3 shows time histories of the vertical accelerations at the locations A and B, obtained with the original and enhanced Substitution methods. It can be observed that the curves corresponding to the same location are virtually identical.

In Figure 4.4(a), the number of iterations needed to achieve convergence with the original algorithm (dashed line) and the enhanced one (solid line) are compared. The number of iterations required for a given tolerance in the original algorithm ranged between 3 and 8, while the range has always been between 1 and 3 for the enhanced algorithm. The periodic pattern of the number of iterations is explained by the oscillatory response of the structure which can be observed in Figure 4.4(b), where the displacements history at location B is superposed to the plot of the number of iterations per time step. The number of required iterations is greater when the displacements are larger.



(a) number of iterations to convergence per time step in original and enhanced algorithms



(b) displacement evolution at point B and corresponding number of iterations

Figure 4.4: Assessment of convergence property.

The fast convergence of the enhanced algorithm can be appreciated in Figure 4.5, where convergence of vertical accelerations at location B is shown over a short period of time between $t = 0.214\ s$ and $t = 0.222\ s$. The blue line represents the solution at convergence, while the square markers denote the solution in the pre-computation phase (iteration 0), which does not take into account the local internal forces. The improvement of precision after the first iteration can be noticed by looking at the circular markers, which are already very close to the converged solution.

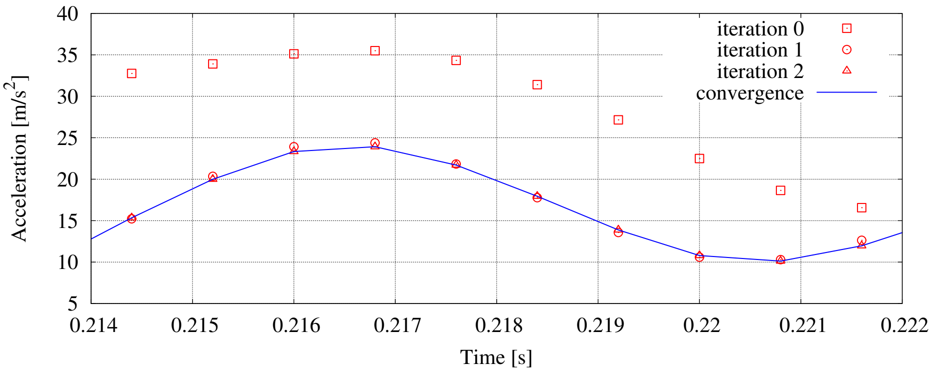


Figure 4.5: Enhanced approach: evolution with iterations of vertical acceleration at location B.

Figure 4.6 allows one to compare the rate of convergence of the two algorithms at different times. The enhanced method exhibits a higher convergence rate, which does not change throughout the history of deformation.

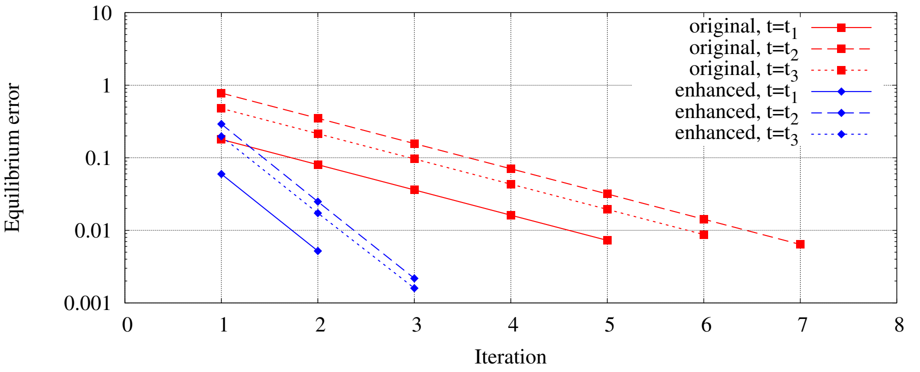


Figure 4.6: Convergence rate of original and enhanced methods at three times: $t_i = \frac{1+i}{6} t_{\text{end}}$ with $i = 1, 2, 3$ and $t_{\text{end}} = 2.5\ s$.

5.2 Local defects and heterogeneities

One of the potential uses of the Substitution method concerns the analysis of a structure which is characterized by local small features. To study this type of situation, the introduction of small defects (either holes or stiff inclusions) is considered in the model just taken previously, shown by the dashed lines which represent defects contours in Figure 4.7. The location S, which will be subsequently used in the presentation of the results, has been singled out in Figure 4.7. The Substitution method allows one to use the same uniform global model chosen in the previous cases, the defects being introduced at the local level only.

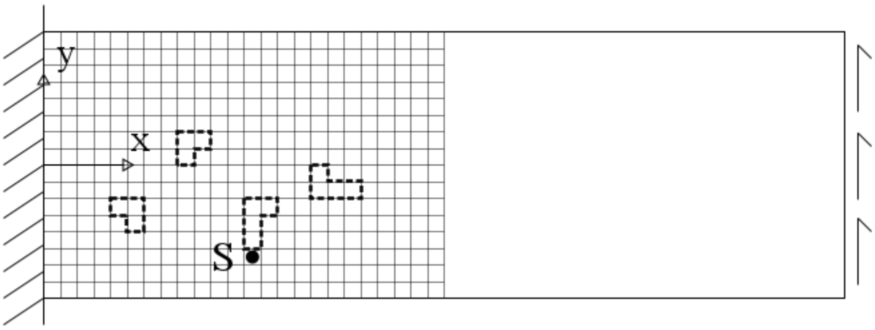


Figure 4.7: Local region with heterogeneities: definition of location S at which the stress solution is monitored.

The introduction of holes does not modify the critical time step of the local model. This is not the case when introducing stiff inclusions, which have been assumed to have a density value equal to the one of the surrounding material, but a ten times larger Young's modulus. For stiff inclusions, this change of material properties leads to a local time step $\Delta t_\ell = \Delta t_g / M = 0.333 \cdot 10^{-4}$ s, with $M = 24 > h_{g/\ell}$.

Figure 4.8 shows that the presence of defects, either holes or stiff inclusions, does not affect the convergence rate. The only minor difference with respect to the homogeneous case concerns the initial error which has increased since the used global model is less representative of the real structural behavior.

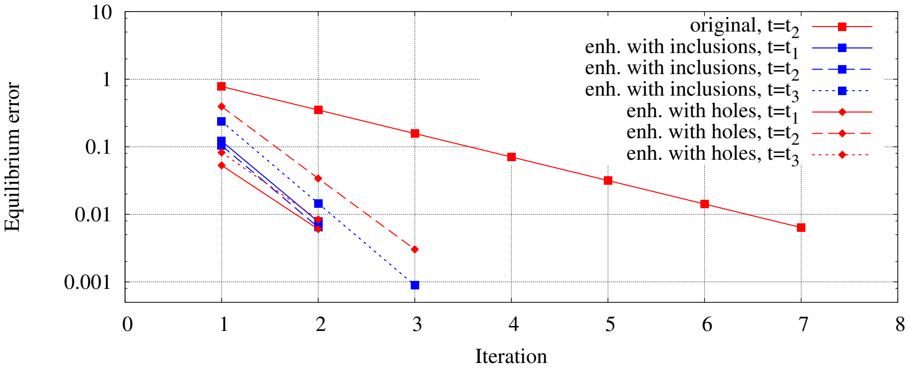
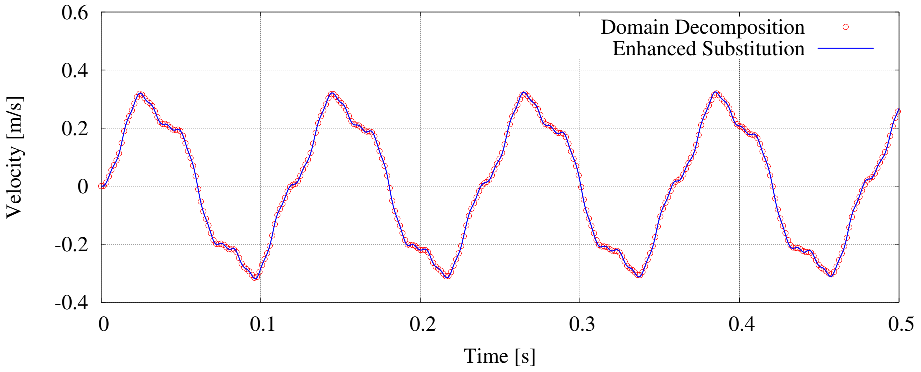
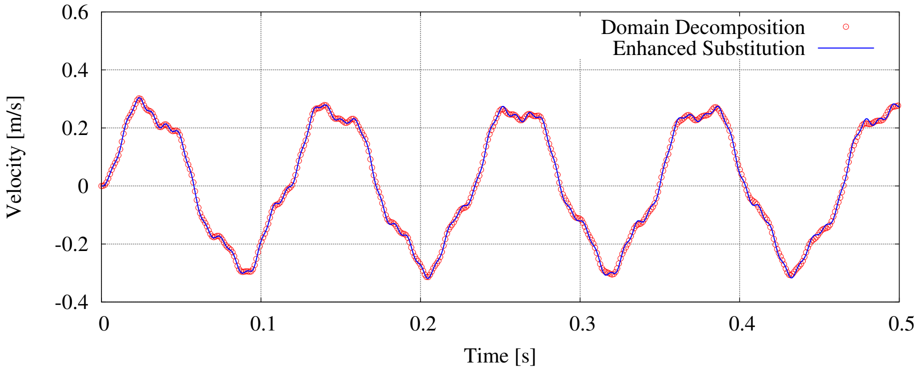


Figure 4.8: Local region with defects: convergence rate at three times $t_i = \frac{1+i}{6} t_{\text{end}}$ with $i = 1, 2, 3$ and $t_{\text{end}} = 2.5$ s, compared to convergence rate of homogeneous problem solved with original method.

Figures 4.9 show the numerical results in terms of vertical velocities at the location B. The solid blue line is the solution obtained with the proposed algorithm while the red circles denote the solution obtained by the application of the Domain Decomposition



(a) analysis with holes



(b) analysis with stiff inclusions

Figure 4.9: Vertical velocities at the location B, obtained with Domain Decomposition with GC algorithm and enhanced Substitution.

method with *GC* algorithm. A good agreement is remarked both for holes and for stiff inclusions.

Now, let σ_x^{st} denote the horizontal stress at the location S in the analogous static solution of the homogeneous problem (i.e. without defects). Figures 4.10(a), for the case with holes, and 4.10(b), for the case with stiff inclusions, show snapshots of the local domain displaying the horizontal stress σ_x at the times when it attains its maximum value. In Figure 4.10(c), the histories of σ_x normalized by σ_x^{st} at the location S are compared with the one obtained without heterogeneities. The expected effects of both holes and inclusions are confirmed in all cases.

5.3 Composite structure with damageable interface

A detailed study of the structural response up to failure often requires taking into account phenomena which spread over multiple spatial and temporal scales. An interesting potential application of the Substitution method is the treatment of this type of problems by introducing local refinements in space and time only when and where needed, avoiding complex implementation issues.

The adaptive introduction of an evolving local region has been considered for the dynamic simulation of delamination. Two 2-dimensional test cases with plane strain assumption are considered:

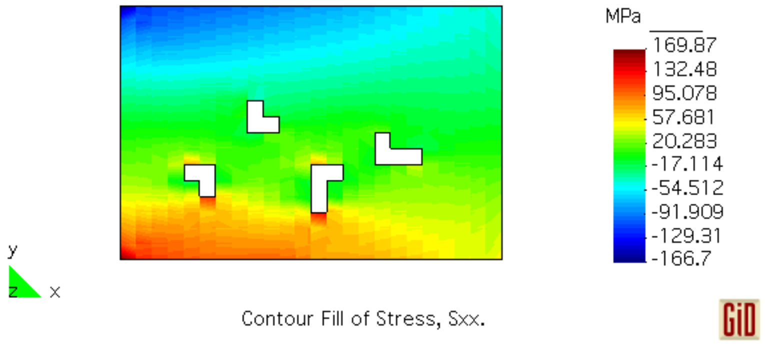
- the End Loaded Split (ELS) test case, described in following Section 5.3.1;
- the Mixed-Mode Bending (MMB) test case, described in following Section 5.3.2.

In both the test cases, a *meso*-scale model is employed in the local analysis, based on two stacked linear elastic plies discretized by 4-node displacement-based quadrilateral elements, connected by cohesive interfaces, discretized by 4-node zero-thickness cohesive elements, conforming with the quadrilateral elements.

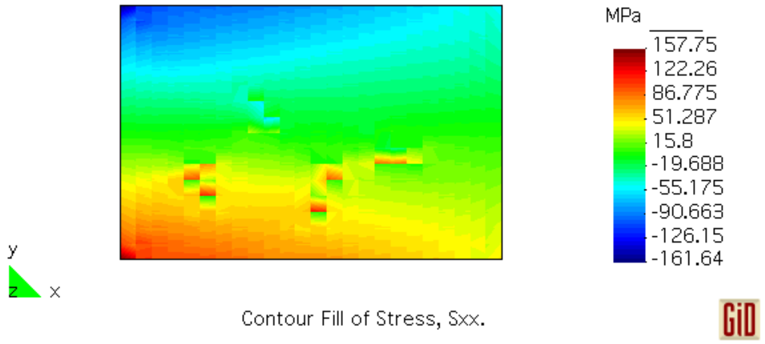
The technology used for the cohesive interface and the associated damage law are described in Appendix A. The damage rate $\dot{d}$ is expressed as inversely proportional to the characteristic time of the fracture process τ_c and with an exponential law function of the isotropic interface stiffness k_0, the interface stress state and the threshold and critical thermodynamical forces Y_0 and Y_c, respectively,.

The material data of the problems are:

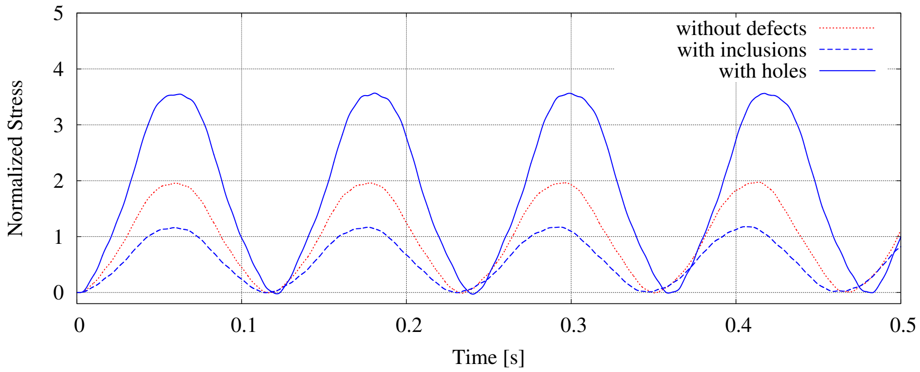
- for the linear elastic plies:
  - density $\rho = 1500\ kg/m^3$;
  - Young's modulus $E = 140 \cdot 10^9\ Pa$;
  - Poisson's ratio $\nu = 0.4$;
- for the cohesive interface:
  - characteristic time $\tau_c = 20 \cdot 10^{-5}\ s$;



(a) stress σ_x distribution over the local domain at the time $t = 0.0616\ s$



(b) stress σ_x distribution over the local domain at the time $t = 0.0592\ s$



(c) stress histories $\sigma_x(t)$ at the location S normalized by the static estimation σ_x^{st}

Figure 4.10: Effects of heterogeneities on the stress solution.

- threshold thermodynamical force $Y_0 = 50 \cdot 10^3$ *Pa*;
- critical thermodynamical force $Y_c = 230 \cdot 10^3$ *Pa*;
- isotropic stiffness $k_0 = 10^{12}$ N/m^3.

A *macro*-scale model is then used in the global coarse analysis. It does not take into account the composite structure and is assumed to be linear elastic with the same material data of the plies. Such model consists of 4-node *EAS-7* quadrilateral elements, formulated in Appendix B and whose integration points are 2×2 Gauss points described in Figure 4.11, featuring a shear stress σ_{xy} constant in the element and a normal stress σ_x constant along the axis x and linearly varying along the axis y.

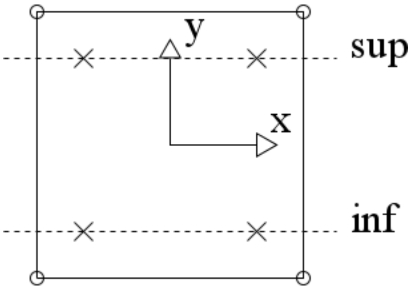


Figure 4.11: Coordinates and quadrature points in the quadrilateral element.

The local adaptive model is composed of assembled elementary local units, each one composed analogously to the reference *meso*-scale model by 4×4 displacement-based quadrilateral elements and 4 cohesive elements located along the cohesive interface. A new unit is added to the left side of the *local* region each time that its interface Γ^h with the *global complementary* region moves to the left. The local problem is solved by making use of the Domain Decomposition method with *GC* algorithm, according to which every elementary local unit is treated as a subdomain of the local region and all the units are analyzed with the same local time step.

The adaptive activation of the local region is governed by a stress criterion in the global analysis. A new local unit is activated when $\mid \sigma_x^{\mathrm{sup}} - \sigma_x^{\mathrm{inf}} \mid /2 > \overline{\sigma}$, where σ_x^{sup} is the σ_x stress component at the two Gauss points above the cohesive interface, while σ_x^{inf} is the corresponding value at the Gauss points below the interface. The threshold $\overline{\sigma}$ has to be tuned in such a way that the cohesive process zone is entirely contained within the *local* region. In the considered example, the interface parameters are such that the process zone extends over the length of several units, so that new units have to be introduced early in the analysis. Three different threshold values have been tested:

- $\overline{\sigma} = 100 \cdot 10^6$ *Pa*, denoted by *act1* as early activation;
- $\overline{\sigma} = 300 \cdot 10^6$ *Pa*, denoted by *act2* as intermediate activation;
- $\overline{\sigma} = 500 \cdot 10^6$ *Pa*, denoted by *act3* as delayed activation.

5.3.1 End Loaded Split test case

As described in Figure 4.12, a structure composed of two linear elastic plies connected by one cohesive interface is loaded at the top right corner.

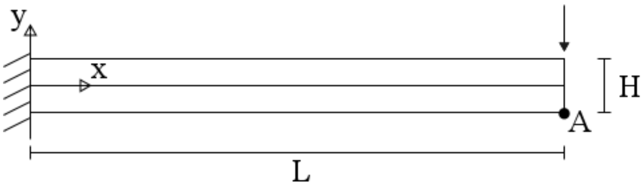


Figure 4.12: End Loaded Split (ELS) test case: sizes, monitored location A and coordinates system.

The data of the problem are:

- length $L = 1\ m$;
- height $H = 0.1\ m$;
- concentrated load $\overline{f} = 10^7\ N/m$ constant in time.

Figure 4.13 describes the uniform mesh $\Omega_r^{h,\text{meso}}$ of the reference monolithic analysis. A corresponding time step $\Delta t_r = 0.5 \cdot 10^{-6}\ s$ is chosen close to the critical estimation.

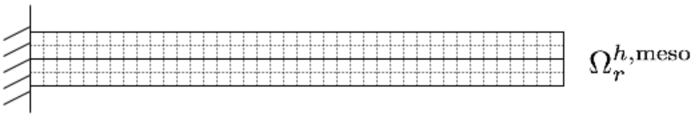


Figure 4.13: ELS reference model: uniform meso-scale mesh.

Utilizing the Substitution method, the coarse global analysis applied to the whole structure is coupled with the adaptive *meso*-scale local region that starts from the right part of the structure and progressively gets bigger in size in order to follow the delamination propagation as illustrated in Figure 4.14. While the global coarse mesh remains the same throughout the analysis, the interface between the *global complementary* and the *local* regions moves to the left as the analysis progresses.

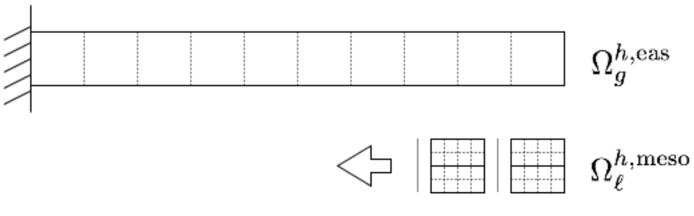


Figure 4.14: ELS Substitution adaptive strategy: fixed global and adaptive local models.

The time is discretized with:

- $\Delta t_g = 2 \cdot 10^{-6}\ s$ as global time step,
- $\Delta t_\ell = \Delta t_g / M = 0.5 \cdot 10^{-6}\ s$ as local time step,

with $M = h_{g/\ell} = 4$.

The interface Γ^h, shared by the *global substitution* and *complementary* regions, is identified by the red nodes in Figure 4.15 and is progressively moved to the left; on the other hand, the blue nodes, inside the *local* and *global complementary* regions are not involved in the iterative process, but they are characterized by *direct substitution* as introduced in Section 5.1 of Chapter 3.

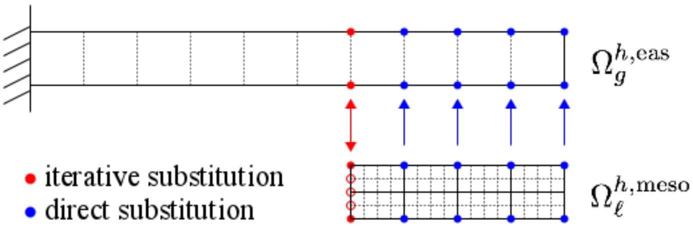


Figure 4.15: ELS Substitution coupling strategy: iterative and direct substitution.

Figure 4.16 shows the numerical results in terms of the vertical velocity of point A. The dashed red line denotes the reference solution. The blue lines indicate the solution with the adaptive Substitution method: the solid and dashed lines denote respectively the analysis with early activation (*act1*) and the analysis with delayed activation (*act3*). As it can be derived from the plot, the particular type of loading, inducing normal stresses in the interface between the plies, activates the structure highest eigenfrequencies due to the high value of the interface stiffness k_0. This aspect is emphasized in the curves relevant to the Substitution method, due to reflections produced by the interface between the *local* and *global complementary* regions.

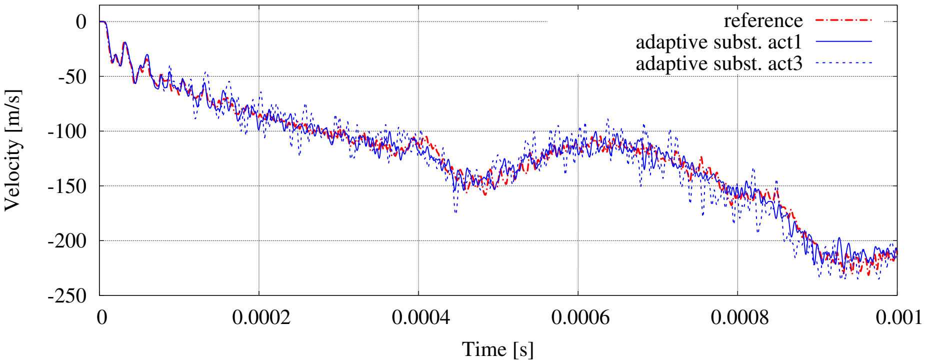


Figure 4.16: ELS test case: comparison of vertical velocities at the location A between reference and adaptive Substitution method, the latter with two different activation thresholds.

The better accuracy of the solution plotted with the solid blue line is due to the lower activation value that increases the size of the local analysis faster. Figure 4.17 shows the number of activated local subdomains during the analysis. The three different activation values are compared: the solid line with *act1*, the dashed line with *act2* and the dotted line with *act3*. The computational gain in comparison with the reference analysis is constituted by the number of subdomains that are not activated during the analysis, i.e. the area above the curves in Figure 4.17. On the other hand, the number of iterations in the Substitution method plays in favor of the reference analysis, that in this particular case is convenient.

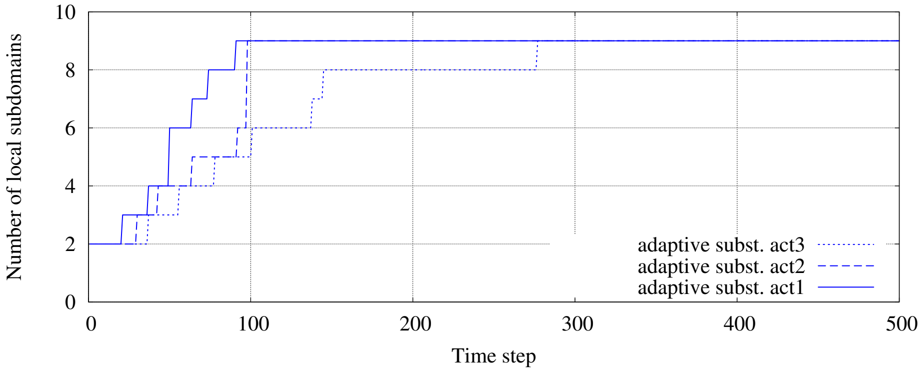


Figure 4.17: Number of activated local subdomains during the analysis.

The predicted position of the delamination front strongly depends on the adopted activation criterion as well, as can be seen in Figure 4.18. Three different delamination fronts, obtained using the three different activation thresholds, are compared with the reference solution (solid line): the dashed line corresponds to the activation threshold *act1*, the short-dashed line to *act2* and the dotted line to *act3*. In all cases, the cohesive process zone is always entirely contained in the *local* region. The Substitution results converge to the reference one s with the anticipation of the the activation process.

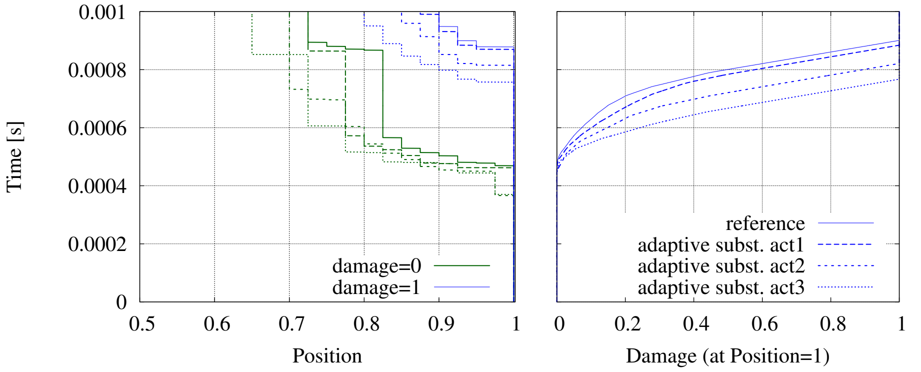


Figure 4.18: ELS test case: delamination process zone and damage evolution for different activation thresholds for local region expansion.

Figure 4.19 shows a sequence of snapshots of the analysis. Starting from a *local* region discretized by means of two units at $t = 0$ s, it progressively increases its size in order to include the delamination front and process zone tip, arriving at a final size of nine subdomains at $t_{\text{end}} = 0.001$ s.

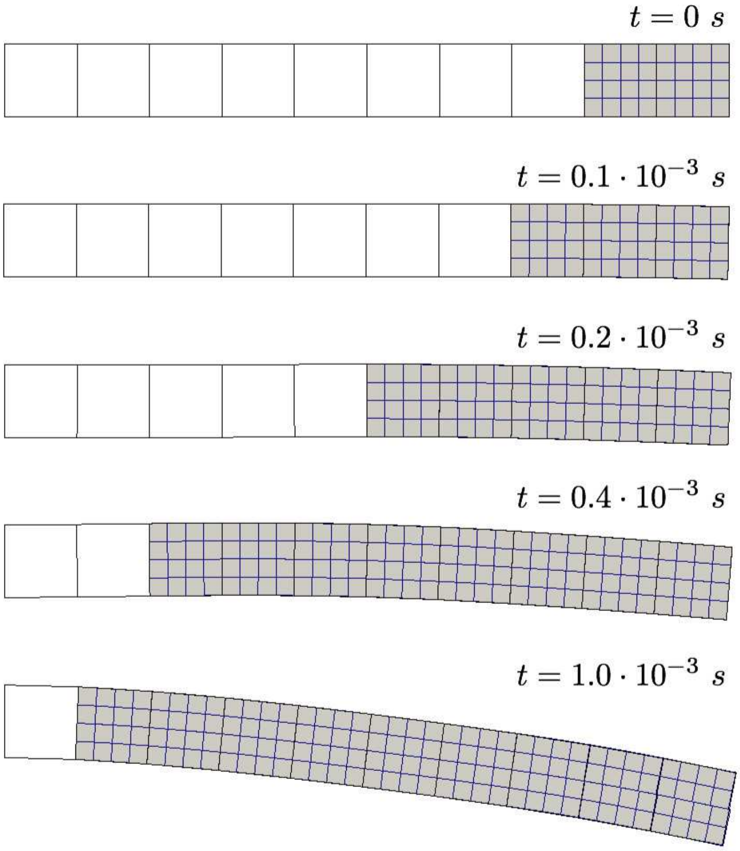


Figure 4.19: ELS test case (act3): analysis snapshots at five representative times.

Figure 4.20 shows the convergence rate at three representative times: one can observe that it is comparable to the one obtained in the previous examples for a purely elastic material behavior. However, in this case the initial error is higher: this is due to the fact that the error associated to the global coarse model is higher than in the previous examples.

The decrease in the number of iterations during the analysis is shown in Figure 4.21: the original algorithm (dotted line) required around 8 iterations to converge, though the proposed algorithm converges with 3 iterations maximum.

5.3.2 Mixed-Mode Bending test case

As described in Figure 4.22, a structure composed of two linear elastic plies connected by one cohesive interface is simply supported and loaded at the middle of the top surface.

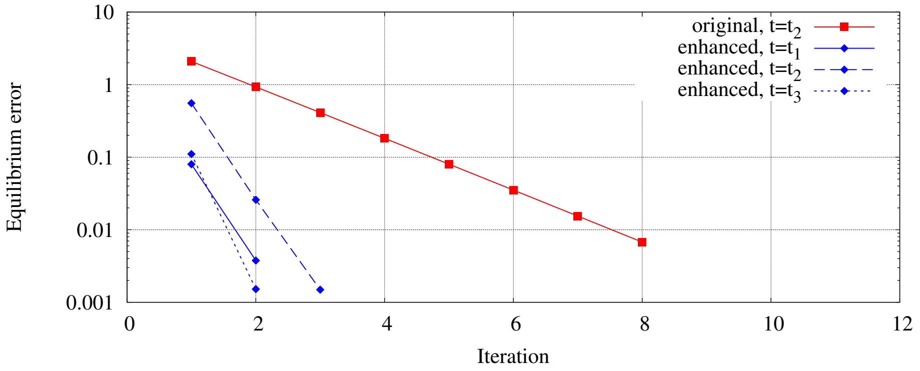


Figure 4.20: ELS test case (act3): convergence rate at three times $t_i = \frac{1+i}{6} t_{\text{end}}$ with $i = 1, 2, 3$ and $t_{\text{end}} = 0.001\ s$.

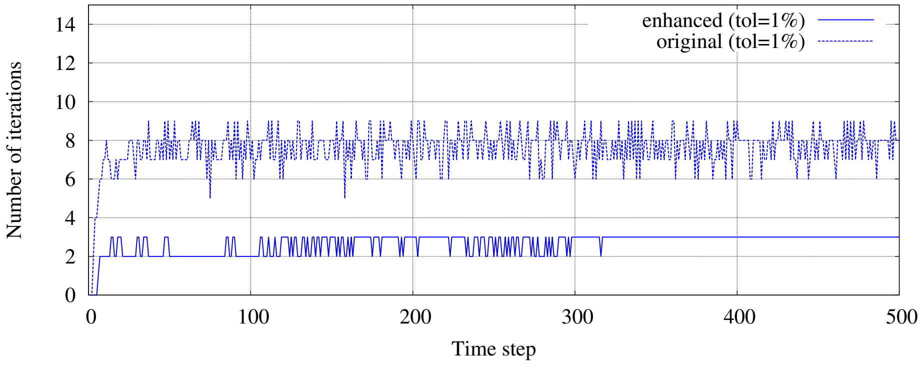


Figure 4.21: ELS test case (act3): gain in terms of number of iterations.

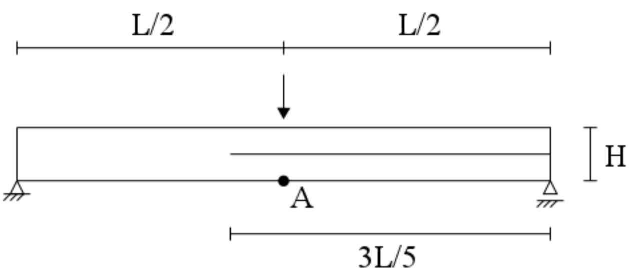


Figure 4.22: Mixed-Mode Bending (MMB) test case: sizes and monitored location.

The data of the problem are:

- length $L = 1\ m$;
- height $H = 0.1\ m$;
- concentrated load $\overline{f} = 1.25 \cdot 10^7\ N/m$ constant in time.

Figure 4.23 describes the uniform mesh $\Omega_r^{h,\text{meso}}$ of the reference monolithic analysis, taking a corresponding time step $\Delta t_r = 0.5 \cdot 10^{-6}\ s$, close to the critical estimation.

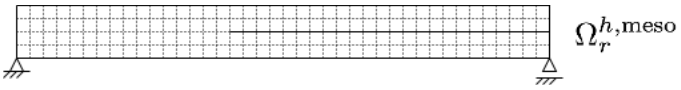


Figure 4.23: MMB reference model: uniform meso-scale mesh.

In this case, the Substitution method couples a coarse global analysis of the whole structure with a local region that starts from the right part of the structure as in Figure 4.24 and is then activated when necessary in order to cover the cohesive interface.

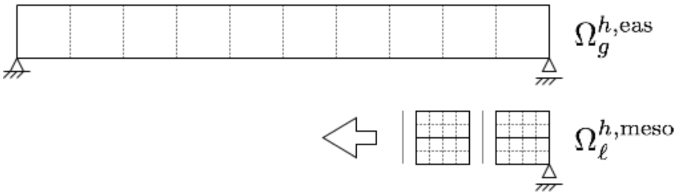


Figure 4.24: MMB Substitution adaptive strategy: fixed global and adaptive local models.

The time is discretized with:

- $\Delta t_g = 2 \cdot 10^{-6}\ s$ as global time step,
- $\Delta t_\ell = \Delta t_g / M = 0.5 \cdot 10^{-6}\ s$ as local time step,

with $M = h_{g/\ell} = 4$.

Once again, the interface Γ^h, shared by the boundaries of the *global substitution* and *complementary* regions, is defined by the red nodes as in Figure 4.25 and progressively moves. The blue nodes, inside the *local* and *global complementary* regions are not involved in the iterative process, but they are subject to *direct substitution* as introduced in Section 5.1 of Chapter 3.

Figure 4.26 shows the numerical results in terms of the vertical velocity of point A. The dashed red line denotes the reference solution. The blue lines indicate the solution with the adaptive Substitution method: the solid and dashed lines denote respectively early activation (*act1*) and delayed activation (*act3*). As it can be exerted from the plot, the peculiar type of loading, inducing normal stresses in the interface between the plies,

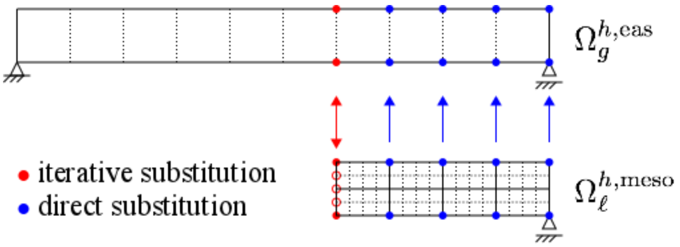


Figure 4.25: MMB Substitution coupling strategy: iterative and direct substitution.

activates the structure highest eigenfrequencies due to the high value of the interface stiffness k_0. This aspect is emphasized in the curves relevant to the Substitution method, due to reflections produced by the interface between the local and complementary regions. The same type of spurious high frequency oscillations would have been obtained with any other Domain Decomposition method to couple the scales.

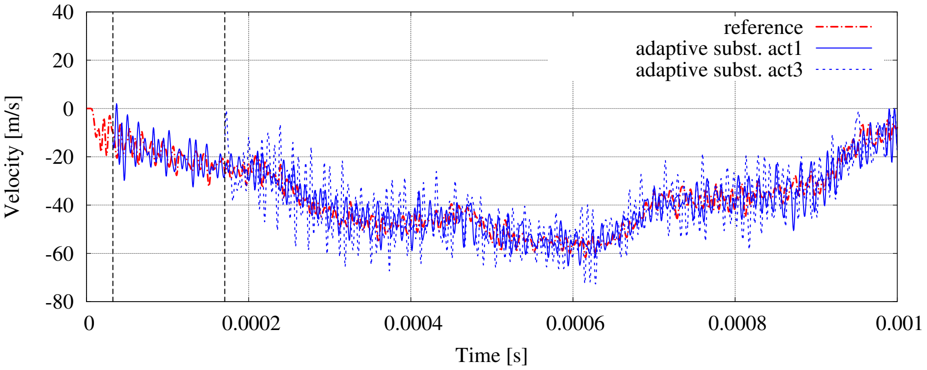


Figure 4.26: MMB test case: vertical velocity at the location A between reference and adaptive Substitution method, with two different activation thresholds.

The predicted position of the delamination front strongly depends as well on the adopted activation criterion, as can be seen in Figure 4.27, where three different delamination fronts, obtained using the three different activation thresholds, are compared with the reference solution (solid line): the dashed line corresponds to the activation threshold *act1*, the short-dashed line to *act2* and the dotted line to *act3*. In all cases, the cohesive process zone is always entirely contained in the local region and, once again, curves converge to the reference solution anticipating the activation of the local subdomains.

Figure 4.28 shows a sequence of snapshots of the analysis. Starting from a local region discretized by means of two units at $t = 0$ s, it instantaneously covers all the cohesive interface when needed, arriving at a final size of six subdomains at $t_{\text{end}} = 0.001$ s.

The number of iterations during the analysis using the enhanced iterative algorithm is shown in Figure 4.29 with at least 3 iterations required to reach the convergence.

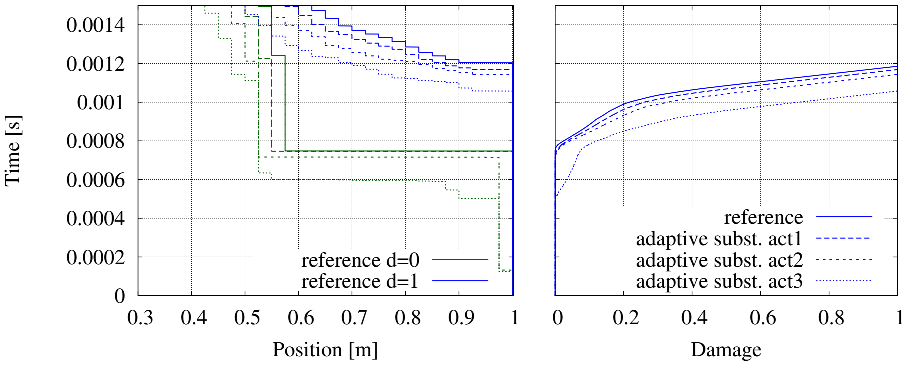


Figure 4.27: MMB test case: delamination process zone and damage evolution for different activation thresholds for local region expansion.

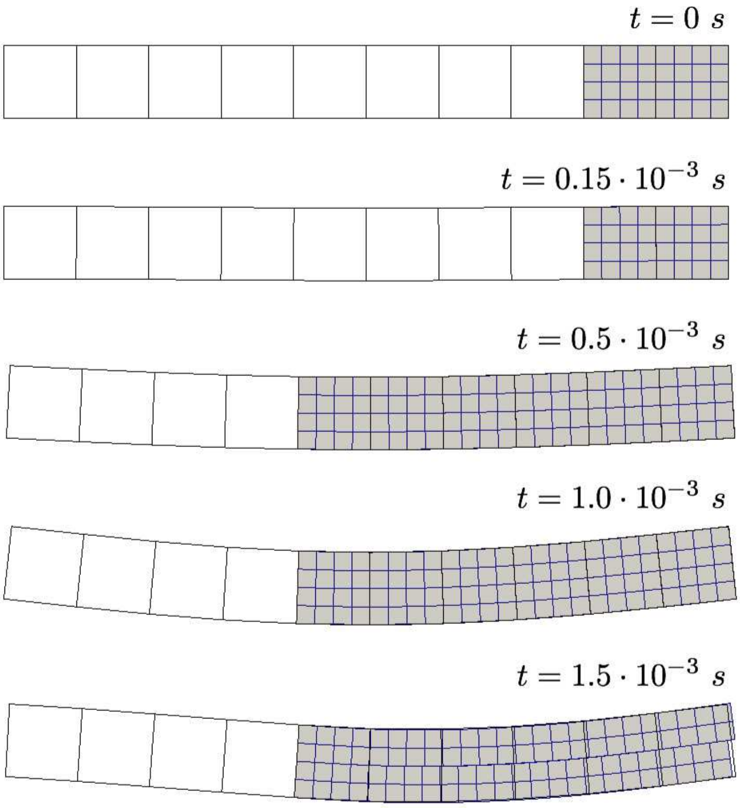


Figure 4.28: MMB test case: analysis snapshots at five representative times.

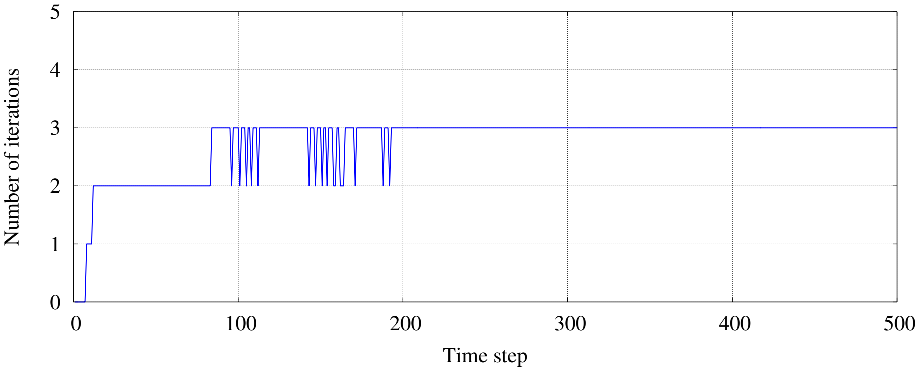


Figure 4.29: Number of iterations in the MMB test case.

6 Conclusions

The primary goal of the study in this Chapter has been the one of improving the iterative scheme of the Substitution method proposed in Chapter 3. This has been achieved thanks to the properties of combination between Central Difference scheme and cubic Hermitian interpolation in time. A consistent coefficients matrix has been derived: it allows one to reduce the number of iterations to a level which is now considered as acceptable for applications of practical interest.

The delamination problem in Section 5.3 has shown the potential applicability of the approach to evolutive nonlinear problems and opens the way to interesting future developments. Nevertheless, the elements used for the global model are quite different from the ones used in the local region as, for example, their stiffness does not coincide with the condensed stiffness of the underlying local model.

In the applications for composites, the adaptive refinement procedure activates the local analysis where delamination could occur and the process zone could be activated. On the other hand, the local analysis becomes unnecessary when the delamination is well defined and the process zone has moved away. A useful enhancement would then consist in the possibility to integrate a de-refinement procedure to the substitution strategy, deactivating the local region and defining an equivalent *macro*-scale model in the global analysis with a weakened constitutive behavior, in the line of what proposed, e.g. in [Sukumar et al., 2000], [Oliver et al., 2003] and [Armero and Linder, 2009], where a crack was embedded on top of a given Finite Element model.

Last, but not least, further investigations could interest the research of a time downscaling operator alternative to the cubic Hermitian interpolation introduced in Section 3.4 of Chapter 3 and which could be able to ensure the displacements continuity over the interface as well as continuity of velocities and accelerations. Such property could allow one to improve stability and accuracy of the coupling scheme and to estimate the local internal forces in the iterative scheme.

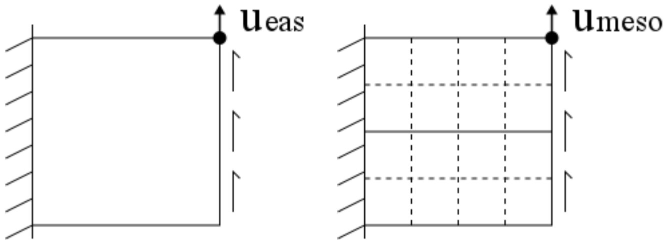
6.1 Investigation towards weakening procedures

Several techniques could enhance the behavior of the global model in order to approach the one of the local model. Such enhancement could be employed to reduce the error in the coupling, e.g. taking account of the defects as in Section 5.2 or of the *mode-2* mechanism of composite structures in Section 5.3.

Considering the ELS test case in Section 5.3.1, due to the complexity of the *meso*-scale model involving contact over the cohesive elements, a simplified coarsening technique is here proposed: fixing the Poisson's ratio $\overline{\nu}$, an equivalent Young's modulus $\tilde{E}$ is iteratively searched so that, considering the *macro*- and *meso*-scale (with linear cohesive interfaces) models shown in Figure 4.30, the solutions of:

$$\mathbf{M}\ddot{\mathbf{u}} = \mathbf{f}_{\text{ext}} - \mathbf{f}_{\text{int}} - \mathbf{f}_{\text{bulk}} \tag{4.22}$$

at the top right corners are equal at $t \to +\infty$, with $\mathbf{f}_{\text{bulk}}$ as artificial forces for damping out the oscillations. In this way, the global analysis would be carried out with an equivalent stiffness that contemplates the refined mesh and the composite structure of the local analysis. Note that such *a priori* calculation strongly depends on arbitrary boundary condition and load choices and is not convenient for general purposes. Nevertheless, these results could be useful to validate further investigations in more advanced weakened models.



(a) global element (b) local subdomain

Figure 4.30: Models used in the weakening procedure.

Re-performing the analysis in Section 5.3.1 with the new weakened global model, the new convergence rates are shown in Figure 4.31. The solid red line with squares denotes the convergence rate of the original algorithm in Chapter 3, the red lines with the rhombuses denote the convergence rates of the enhanced algorithm without weakening as in Section 5.3 and the blue lines indicate the convergence rates of the enhanced algorithm taking $\tilde{E} = 0.737 \cdot E$ in the global analysis, as a result of the weakening procedure assuming linear elastic cohesive interfaces. Blue lines start from a lower position and tend to be slightly faster.

6.2 Investigation towards de-refinement procedures

The same procedure described in Section 6.1 could be applied to the global model in the de-refinement of the analysis, as illustrated in Figure 4.32. An equivalent Young's

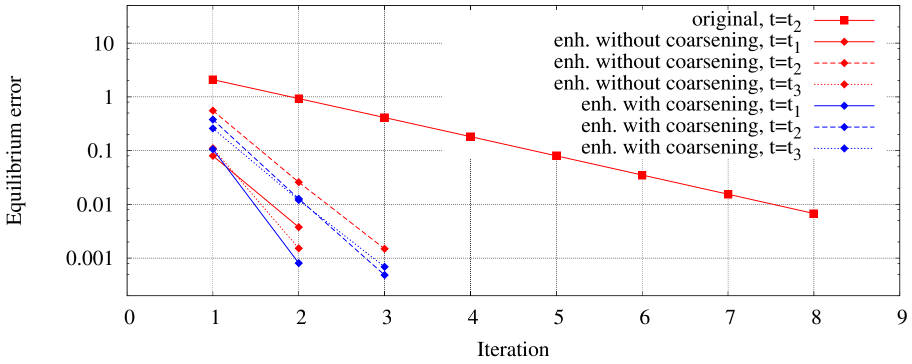


Figure 4.31: Convergence rates at the three times $t_i = \frac{1+i}{6} t_{\text{end}}$ ($i = 1,2,3$) in the ELS test case with or without weakening procedure.

modulus for the *macro*-scale model is reduced to $\tilde{E} = 0.431 \cdot E$, out of the solutions of Equation (4.22) considering a *meso*-scale model in Figure 4.30 without cohesive interfaces (because completely damaged). A new region is so defined in the global analysis as *weakened global complementary* region $\tilde{\Omega}^h_{gc}$.

Note that the constraint choice over the new interface between *weakened global complementary* $\tilde{\Omega}^h_{gc}$ and *local* Ω^h_ℓ regions is crucial. The activated *mode-2* mechanism in the local analysis does not allow to constrain all the nodes over the interface with a linear interpolation in space. Here, only external nodes of the interface are constrained.

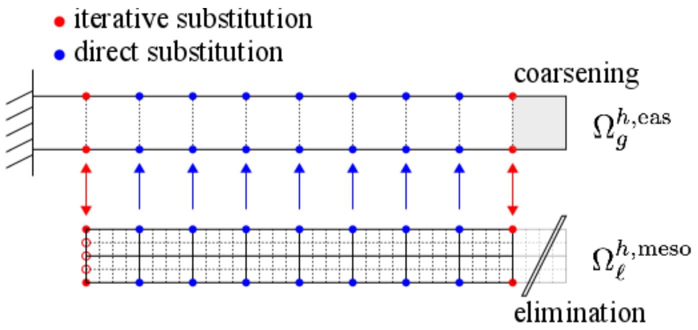


Figure 4.32: De-refinement strategy with the coarsening of the meso-scale model to the macro-scale one.

Figure 4.33 shows a snapshot of the end of the analysis in which the de-refinement is activated, because all the cohesive elements of the local subdomain at the right-hand side of the structure are eliminated because of the fill damage.

6.3 Investigation towards the displacements continuity: quintic Hermitian interpolation

The quintic Hermitian interpolation $\Pi^5_t({}^n\odot, {}^n\square, {}^n\triangle, {}^{n+1}\odot, {}^{n+1}\square, {}^{n+1}\triangle)$ is here tested in order to obtain the displacements continuity, using global displacements, velocities and

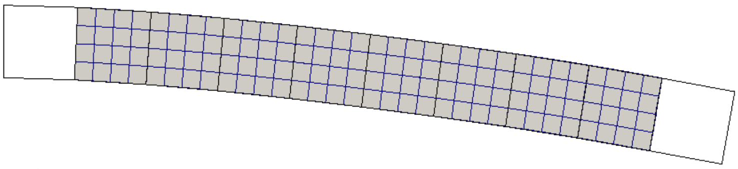


Figure 4.33: Snapshot at the end of the analysis with de-refinement at the right side of the structure.

accelerations in the interpolation, so that:

$$\dot{\mathbf{U}}_{\ell,\Gamma}|_{[t^n,t^{n+1}]} = \Pi_h^1 \Pi_t^5 \left({}^n\mathbf{U}_{g,\Gamma}, {}^n\mathbf{V}_{g,\Gamma}, {}^n\mathbf{A}_{g,\Gamma}, {}^{n+1}\mathbf{U}_{g,\Gamma}, {}^{n+1}\mathbf{V}_{g,\Gamma}, {}^{n+1}\mathbf{A}_{g,\Gamma} \right) \tag{4.23}$$

that, introducing the intrinsic local time $s(t)$ and 6 quintic polynomials $\psi_i(s)$, becomes:

$$\begin{aligned} \dot{\mathbf{U}}_{\ell,\Gamma}|_{[t^n,t^{n+1}]} = {} & \Pi_h^1 \frac{{}^n\mathbf{U}_{g,\Gamma}}{\Delta t_g} \psi_1(s) + \Pi_h^1\, {}^n\mathbf{V}_{g,\Gamma} \psi_2(s) + \Pi_h^1\, {}^n\mathbf{A}_{g,\Gamma} \Delta t_g \psi_3(s) + \\ & + \Pi_h^1 \frac{{}^{n+1}\mathbf{U}_{g,\Gamma}}{\Delta t_g} \psi_4(s) + \Pi_h^1\, {}^{n+1}\mathbf{V}_{g,\Gamma} \psi_5(s) + \Pi_h^1\, {}^{n+1}\mathbf{A}_{g,\Gamma} \Delta t_g \psi_6(s). \end{aligned} \tag{4.24}$$

The 6 quintic polynomials $\psi_i(s)$ with $i = 1, \ldots, 6$ are obtained imposing:

$$\psi_i(0) = \delta_{i1}, \quad \dot{\psi}_i(0) = \delta_{i2}, \quad \ddot{\psi}_i(0) = \delta_{i3}, \quad \psi_i(1) = \delta_{i4}, \quad \dot{\psi}_i(1) = \delta_{i5}, \quad \ddot{\psi}_i(1) = \delta_{i6} \qquad i = 1, \ldots 6 \tag{4.25}$$

so that:

$$\begin{aligned} \dot{\mathbf{U}}_{\ell,\Gamma}|_{[t^n,t^{n+1}]} = {} & \Pi_h^1 \frac{{}^n\mathbf{U}_{g,\Gamma}}{\Delta t_g} \left(-30s^2 + 60s^3 - 30s^4\right) + \\ & + \Pi_h^1\, {}^n\mathbf{V}_{g,\Gamma} \left(1 - 18s^2 + 32s^3 - 15s^4\right) + \Pi_h^1\, {}^n\mathbf{A}_{g,\Gamma} \Delta t_g \left(s - 9s^2/2 + 6s^3 - 5s^4/2\right) + \\ & + \Pi_h^1 \frac{{}^{n+1}\mathbf{U}_{g,\Gamma}}{\Delta t_g} \left(30s^2 - 60s^3 + 30s^4\right) + \Pi_h^1\, {}^{n+1}\mathbf{V}_{g,\Gamma} \left(-12s^2 + 28s^3 - 15s^4\right) + \\ & + \Pi_h^1\, {}^{n+1}\mathbf{A}_{g,\Gamma} \Delta t_g \left(-3s^2/2 - 4s^3 + 5s^4/2\right). \end{aligned} \tag{4.26}$$

The constraint has been enriched. Nevertheless, compatibility with the Central Difference scheme in the local analysis is not satisfied. Considering the velocities constraint in Equation (4.24), its derivation in time and taking $s(t^{m+1})$ and $s(t^m)$, the local Central Difference scheme that needs to be satisfied is:

$$^{m+1}\mathbf{V}_{\ell,\Gamma} - {}^{m}\mathbf{V}_{\ell,\Gamma} = \left({}^{m+1}\mathbf{A}_{\ell,\Gamma} + {}^{m}\mathbf{A}_{\ell,\Gamma}\right)\frac{\Delta t_\ell}{2}. \tag{4.27}$$

The left-hand side of Equation (4.27) becomes:

$$\begin{aligned} &^{m+1}\mathbf{V}_{\ell,\Gamma} - {}^{m}\mathbf{V}_{\ell,\Gamma} = {}^{n}\mathbf{A}_{g,\Gamma}\frac{\Delta t_g}{M} - \left({}^{n+1}\mathbf{A}_{g,\Gamma} - {}^{n}\mathbf{A}_{g,\Gamma}\right)\frac{\Delta t_g}{M}\cdot\overline{\psi}_m^{\text{lhs}} \\ &\overline{\psi}_m^{\text{lhs}} = (2m+1)\frac{9}{2M} - \left(3m^2+3m+1\right)\frac{10}{M^2} + \left(4m^3+6m^2+4m+1\right)\frac{5}{M^3} \end{aligned} \tag{4.28}$$

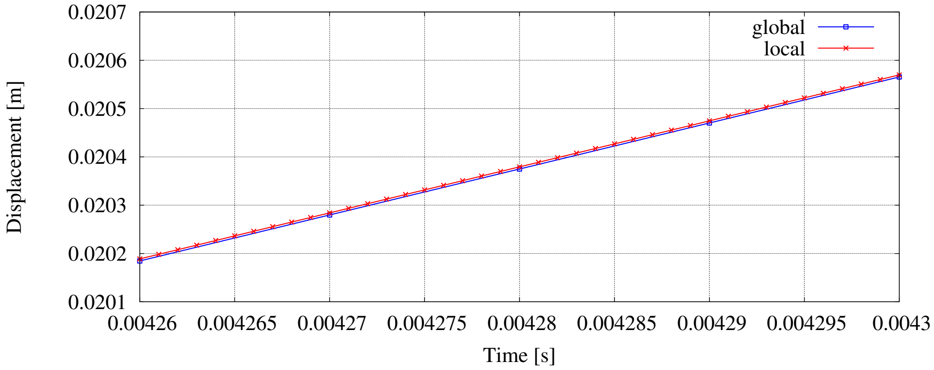
whereas the right-hand side reads:

$$\begin{aligned} &\left({}^{m+1}\mathbf{A}_{\ell,\Gamma} + {}^{m}\mathbf{A}_{\ell,\Gamma}\right)\frac{\Delta t_\ell}{2} = {}^{n}\mathbf{A}_{g,\Gamma}\frac{\Delta t_g}{M} - \left({}^{n+1}\mathbf{A}_{g,\Gamma} - {}^{n}\mathbf{A}_{g,\Gamma}\right)\frac{\Delta t_g}{M}\cdot\overline{\psi}_m^{\text{rhs}}, \\ &\overline{\psi}_m^{\text{rhs}} = (2m+1)\frac{9}{2M} - \left(2m^2+2m+1\right)\frac{15}{M^2} + \left(2m^3+3m^2+3m+1\right)\frac{10}{M^3}. \end{aligned} \tag{4.29}$$

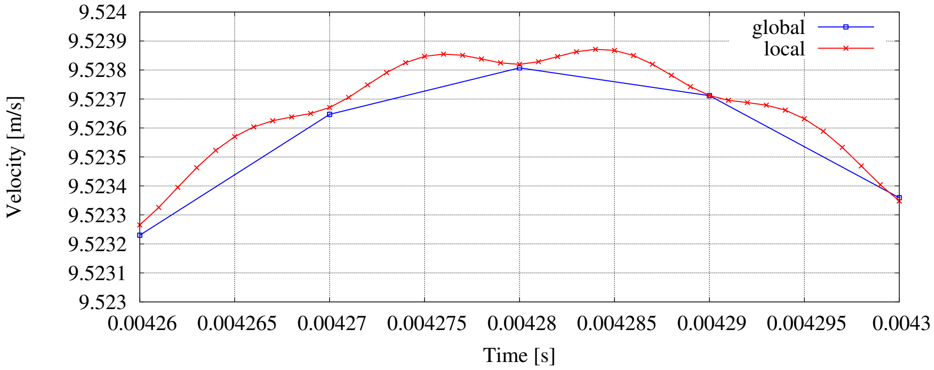
Equation (4.27) turns out to be not satisfied and, subtracting its right-hand side to the left-hand one, one obtains:

$$\overline{\psi}_m^{\text{lhs}} - \overline{\psi}_m^{\text{rhs}} = \frac{5}{M^3}\left(2m - M + 1\right). \tag{4.30}$$

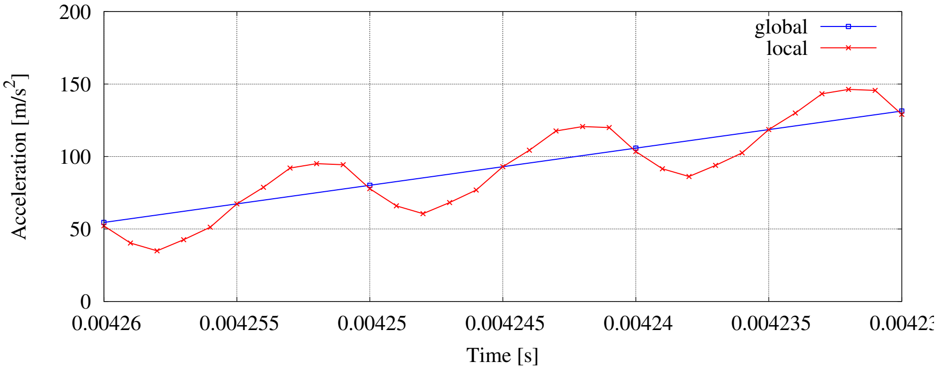
For the spring test example of Figure 3.8 in Section 3.4 of Chapter 3, the global and local quantities are always very close to each other, but not exactly continuous at each global time step, as shown in Figures 4.34.



(a) displacements



(b) velocities



(c) accelerations

Figure 4.34: Kinematic constraint with a cubic Hermitian interpolation in time and applying the Central Difference scheme for global and local solutions and M=10.

Conclusions and prospects

The general topic of the thesis is the extension of non-intrusive coupling techniques, previously developed in statics, to the case of explicit dynamics. One of the objectives of these techniques is the capability to locally enrich existing industrial Finite Element models with no modification of industrial mesh and solver. These methods have recently gained some attention by industrial groups and are essential features of the ANR program ICARE, involving Airbus and EDF, and of other programs, involving SAFRAN. A particular aim of the method in dynamics would be the possibility to perform adaptive modeling or mesh refinement without modifying the global mesh; a target being the coupling of a *macro*-scale model with a *meso*-scale one in order to deal with propagation of large delamination.

Nevertheless, the extension of a substitution approach, where a local model is iteratively substituted within a global analysis, raised several questions in dynamics. Notably in the case of explicit dynamics, stability and efficiency issues make the feasibility and applicability of such approach *a priori* doubtful. This work has been therefore mainly devoted to these two academic aspects: the feasibility of a global-local coupling technique in explicit dynamics and the applicability for industrial purposes with efficiency targets.

In the field of non-intrusive coupling techniques, the global-local algorithm that has been implemented as output of this thesis can be more accurately defined as weakly-intrusive, because the industrial mesh is not modified, but the global solution procedure needs to be updated inside the industrial code because of efficiency issues.

The coupling technique has been implemented in Matlab prototypes and verified in comparison with the non-overlapping Domain Decomposition FETI approaches.

A preliminary stability study has been conducted by the application of the energy method for different time scales. An energy-preserving Substitution method has been developed, leading nevertheless to computationally expensive procedures in case of nonlinearities and adaptive time stepping.

A particular attention has been then dedicated to the improvement of the iterative scheme, in order to meet the efficiency targets for industrial applications. For this purpose, the equality between global and local accelerations, coming from the combination of the Central Difference schemes with the cubic Hermitian interpolation, has allowed for the use of a new mass operator and a new iterative procedure more performant than the original one. Examples with localized defects, such as holes or stiff inclusions, and with nonlinearities have been analyzed, with a first adaptive prototype to follow delamination propagations. The study of such adaptive scheme raised additional difficulties to be investigated.

Last, but not least, the implementation inside the commercial software Abaqus for truss applications has been engaged as a preliminary test of the method inside an extended code architecture. A brief description of the developments is presented in Appendix C. Such implementation as two-way coupling submodeling-like capability via the Co-Simulation technique has been a test for the intrusivity of the model, especially concerning the coupling between different time scales. In particular, the need to go back in time in order to re-iterate the local analysis has required specific code implementations. The extension to solid and shell models and to the existing features is necessary for dealing with industrial problems.

Future developments

In parallel with the extension of the Substitution method inside Abaqus to the solid and shell models and to the existing capabilities of the software, further research studies involve:

- development of a weakened *macro*-scale model to represent the delaminated behavior in the global analysis;
- treatment of contact problems: the easier idea consists in simulating the contact in the global analysis and mapping equivalent nodal forces in the local analysis; more accurate approaches could involve the computation in the local analysis with the use of opportune 2-level operators as the ones introduced for static problems in [Nour-Omid and Wriggers, 1986];
- treatment of non-conforming global-local meshes, considering global-local interfaces that are not aligned with the global elements boundaries; special interpolation functions are for instance proposed in [Biabanaki et al., 2014] considering polygonal elements for large deformations contact problems;
- treatment of wave-reflection over the global-local interface, as for instance in [To and Li, 2005], and use of artificial dissipative techniques alternative to bulk viscosity techniques, introduced for fluid dynamics in [VonNeumann and Richtmyer, 1950] and [Landshoff, 1955] and usually employed in explicit dynamics of structures;
- extension of the coupling technique in order to deal with shell formulations in the global analysis;
- extension of the coupling technique to implicit-explicit couplings as in [Gravouil and Combescure, 2001] and further works.

Appendix A

Meso-scale model for the Matlab prototype

This appendix describes the inter-laminar behavior for stratified composites implemented for the Matlab prototypes.

1 Cohesive element technology

The formulation of the stiffness matrix of a 2-dimensional 4-node cohesive element as developed in [Schellekens and de Borst, 1993a] and [Schellekens and de Borst, 1993b] is here described. Figure A.1 shows a cohesive element in a global coordinate system $\{x,y\}$ with the indexes of the 4 nodes. Note that the nodes 1-3 and 2-4 are perfectly overlapped together to m_1 and m_2, respectively, in the reference configuration. Unit vectors **n** and **t** respectively denote the normal and tangential vectors to the middle-line, ξ indicates the local coordinate along the element.

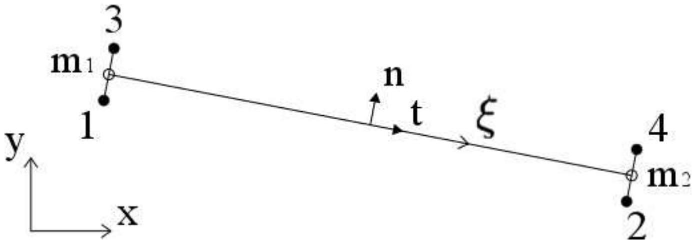


Figure A.1: Cohesive element notation.

The displacement parameters $\mathbf{U}_{\text{coh}}$ associated to the cohesive element (denoted with the subscript “coh”) in the global coordinate system are:

$$\mathbf{U}_{\text{coh}} = \left\{ \; U_{1x}, \;\; U_{1y}, \;\; U_{2x}, \;\; U_{2y}, \;\; U_{3x}, \;\; U_{3y}, \;\; U_{4x}, \;\; U_{4y} \; \right\}^T \tag{A.1}$$

and the displacement jump parameters $\Delta\mathbf{U}_{\text{coh}}$ between adjacent nodes is written as:

$$\Delta\mathbf{U}_{\text{coh}} = \left\{ \; U_{3x}-U_{1x}, \;\; U_{3y}-U_{1y}, \;\; U_{4x}-U_{2x}, \;\; U_{4y}-U_{2y} \; \right\}^T \tag{A.2}$$

considering the nodal indexes and the directions defined in Figure A.1.

The displacement parameters $\mathbf{U}_{\text{coh}}$ and the displacement jump parameters $\Delta\mathbf{U}_{\text{coh}}$ are then linked by the following expression:

$$\begin{aligned} \Delta\mathbf{U}_{\text{coh}} &= \mathbf{\Phi}\mathbf{U}_{\text{coh}} \\ \mathbf{\Phi} &= \left[\; -\mathbf{I}, \;\; \mathbf{I} \; \right] \end{aligned} \tag{A.3}$$

where **I** is the identity matrix.

The linear shape functions N_{m1} and N_{m2} are taken along the middle-line connecting m_1 and m_2, so that:

$$\begin{aligned} \Delta\mathbf{u}_{\text{coh}}(\mathbf{x}) &= \mathbf{H}_{\text{coh}}\Delta\mathbf{U}_{\text{coh}} \\ \mathbf{H}_{\text{coh}} &= \left[\; N_{\text{m1}}\mathbf{I}, \;\; N_{\text{m2}}\mathbf{I} \; \right]. \end{aligned} \tag{A.4}$$

Using the definition in Equation (A.3), the displacement jump field in Equation (A.4) becomes:

$$\Delta\mathbf{u}_{\text{coh}} = \mathbf{H}_{\text{coh}}\mathbf{\Phi}\mathbf{U}_{\text{coh}}. \tag{A.5}$$

The local tangential and normal unit vectors **t** and **n** are obtained by:

$$\begin{aligned}\mathbf{t} &= \frac{1}{J}\left\{\ \partial x/\partial\xi,\quad \partial y/\partial\xi\ \right\}^T \\ \mathbf{n} &= \frac{1}{J}\left\{\ -\partial y/\partial\xi,\quad \partial x/\partial\xi\ \right\}^T\end{aligned} \tag{A.6}$$

where:

$$J = \sqrt{\left(\frac{\partial x}{\partial\xi}\right)^2 + \left(\frac{\partial y}{\partial\xi}\right)^2}. \tag{A.7}$$

Defining the transformation tensor $\mathbf{\Theta}$:

$$\mathbf{\Theta} = \left[\ \mathbf{t},\quad \mathbf{n}\ \right] \tag{A.8}$$

the local displacement jump parameters $\Delta\mathbf{U}_{tn}$ are so defined:

$$\Delta\mathbf{U}_{tn} = \mathbf{\Theta}\mathbf{H}_{\text{coh}}\mathbf{\Phi}\mathbf{U}_{\text{coh}} \tag{A.9}$$

and the elastic energy of a cohesive element $\mathcal{V}_{\text{coh}}$ writes:

$$\mathcal{V}_{\text{coh}} = \frac{1}{2}\int_{\Gamma_{\text{coh}}} \Delta\mathbf{U}_{tn}^T \mathbb{D}_{\text{coh}} \Delta\mathbf{U}_{tn}\, dL. \tag{A.10}$$

The tangent stiffness matrix $\mathbb{K}_{\text{coh}}$ is defined for each cohesive element as:

$$\mathbb{K}_{\text{coh}} = \int_{\Gamma_{\text{coh}}} \mathbf{\Phi}^T \mathbf{H}_{\text{coh}}^T \mathbf{\Theta}^T \mathbb{D}_{\text{coh}} \mathbf{\Theta}\mathbf{H}_{\text{coh}}\mathbf{\Phi} J\, d\xi \tag{A.11}$$

in which $\mathbb{D}_{\text{coh}}$ denotes the constitutive matrix, that writes:

$$\mathbb{D}_{\text{coh}} = \begin{bmatrix} k_t & 0 \\ 0 & k_n \end{bmatrix} \tag{A.12}$$

where k_n and k_t are the normal and tangential stiffnesses, respectively.

2 Interface contact

The contact between plies is modeled with a simplified *penalty-based* approach, in which forces are applied at nodes to contrast inter-penetration. Such forces are computed as:

$$\mathbf{F}_n = k_c < \Delta\mathbf{U}_n >_- \tag{A.13}$$

where k_c depends on the element size and the *penalty scale factor*, n is the normal direction to the middle-line of the cohesive element and $< \square >_-$ denotes the negative part of $\square$.

For the sake of simplicity, friction is neglected.

3 Isotropic damage model

Many damage models have been developed for the *meso*-scale model. An energy-based damage formulation has been introduced in [Allix and Ladevèze, 1992] and [Allix et al., 1995] for the inter-laminar damage modeling. The same theoretical developments have been used in [Ladevèze and LeDantec, 1992] for intra-laminar models. The extension to high-velocity impact loadings is based on the use of a fracture characteristic time parameter introduced in [Ladevèze et al., 2000] and [Allix, 2001] and employed for industrial applications for instance in [Allix and Blanchard, 2006] and [Guimard et al., 2009].

The isotropic damage model application is so described starting from the definition of energy:

$$\mathcal{E}_D = \frac{1}{2}\int_\Gamma \left(\frac{<\sigma_n>_-^2}{k_n} + \frac{<\sigma_n>_+^2}{k_n(1-d)} + \frac{\sigma_t^2}{k_t(1-d)} \right) dL \tag{A.14}$$

where Γ is the length of the middle-line interface, σ_n and σ_t are respectively the normal and tangential stresses and d is the scalar value representing the isotropic damage.

The thermodynamical forces Y_n and Y_t are introduced as derivatives of the energy $\mathcal{E}_D$ in function of the damage scalar value d, so that:

$$Y_n = \frac{1}{2}\frac{<\sigma_n>_+^2}{k_n(1-d)^2}, \qquad Y_t = \frac{1}{2}\frac{\sigma_t^2}{k_t(1-d)^2}. \tag{A.15}$$

Assuming that the thermodynamical forces are strongly coupled and driven by a unique equivalent thermodynamical force $Y(t)$, such that:

$$Y(t) = (Y_n^\alpha + (\gamma Y_t)^\alpha)^{1/\alpha} \tag{A.16}$$

where γ and α are material parameters and assuming, for the sake of simplicity, that $\gamma = \alpha = 1$, so that:

$$Y(t) = \frac{1}{2}\left(\frac{<\sigma_n>_+^2}{k_n(1-d)^2} + \frac{\sigma_t^2}{k_t(1-d)^2} \right), \tag{A.17}$$

the damage evolution law is defined as:

$$\dot{d} = \frac{1}{\tau_c}\left(1 - e^{-<\frac{1}{2}\frac{<\overline{Y}-Y_0>_+}{Y_c-Y_0} - d>_+} \right) \tag{A.18}$$

where d is the damage value, τ_c is the characteristic time of the fracture process, $<\square>_+$ is the positive part of the quantity $\square$, the constant values Y_0 and Y_c are the threshold and critical thermodynamical forces, respectively, and $\overline{Y}$ writes:

$$\overline{Y}(t) = \sup_{\tau\leq t} Y(\tau). \tag{A.19}$$

4 Application to mode-2 mechanism

Figure A.2 shows the test case utilized here to present the cohesive interface technology and the damage model associated. The structure is composed of two stacked bodies with linear elastic behavior, connected by the cohesive interface at mid-height. The bottom edge of the structure is clamped, whereas the top edge is constrained to a given velocity $\overline{\mathbf{v}}$. The plane strain condition is assumed and a mesh of bilinear quadrilateral elements is defined. The cohesive elements along the cohesive interface are 10, conforming with the quadrilateral elements. The stress and damage variables are monitored at the quadrature point S.

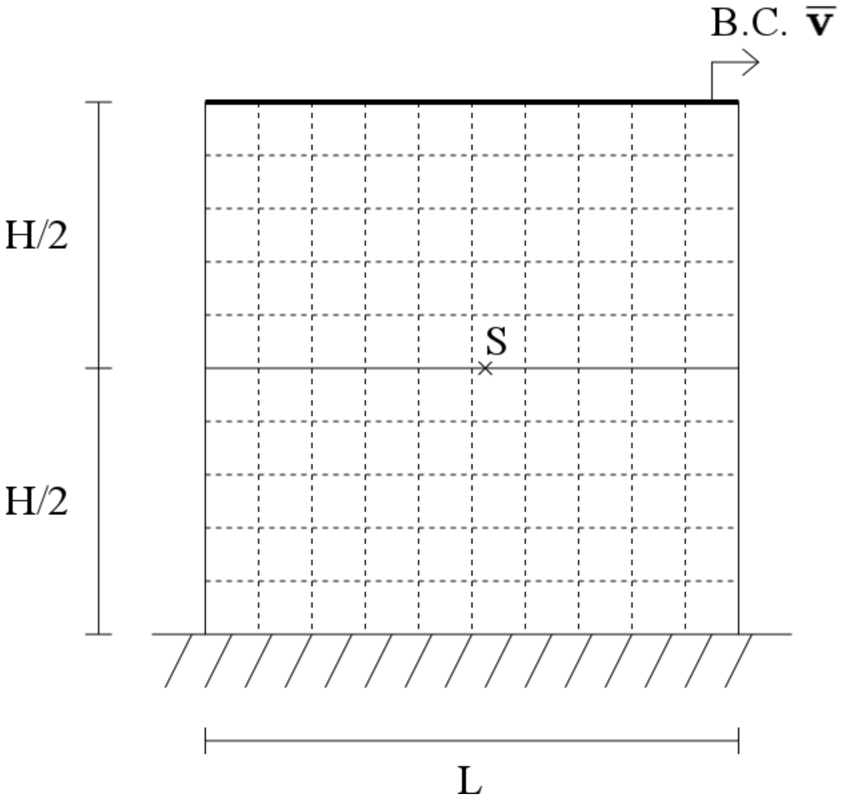


Figure A.2: Structure with cohesive interface at the mid-height.

The problem data are:

- for the linear elastic bodies:
  - density $\rho = 1500\ kg/m^3$;
  - Young's modulus $E = 140 \cdot 10^9\ Pa$;
  - Poisson's ratio $\nu = 0.4$;
- for the cohesive interface:

  - characteristic time $\tau_c = 20 \cdot 10^{-6}\ s$;
  - threshold thermodynamical force $Y_0 = 50 \cdot 10^3\ Pa$;
  - critical thermodynamical force $Y_c = 230 \cdot 10^3\ Pa$;
  - interface stiffness $k_n = k_t = k_0 = 10^{12}\ N/m^3$;

- length $L = 1\ m$;
- height $H = 1\ m$;
- imposed boundary condition $\overline{\mathbf{v}} = \{v_x, v_y\}^T = \{10, 0\}^T\ m/s$ over the top edge.

A time step $\Delta t = 2 \cdot 10^{-6}\ s$ is chosen out of the relations with the mesh size.

Figure A.3 shows the results at the location S in terms of thermodynamical forces $\overline{Y}$ (red line) and damage d (blue line). In accordance with the damage law in Equation (A.18), the damage is activated when $\overline{Y} > Y_0 = 50 \cdot 10^3\ Pa$ and follows an exponential law.

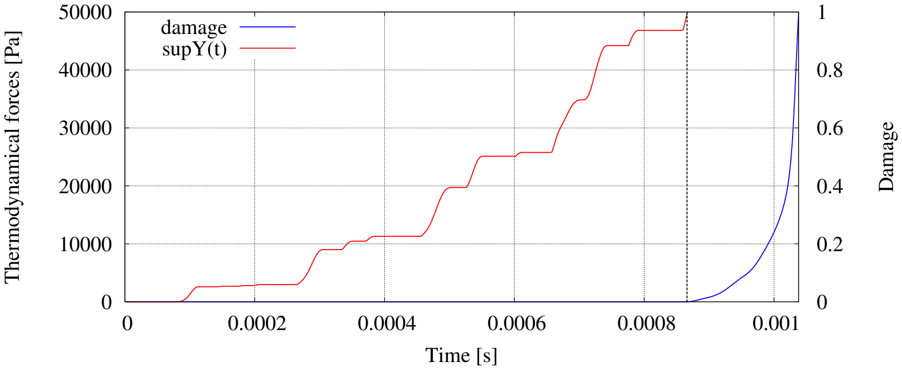


Figure A.3: Thermodynamical forces and damage at the location S.

Figure A.4 shows the results at the location S in terms of tangential stresses, whereas the normal stresses are negligible because of the nature of the boundary conditions. The damage is initiated when $\sigma_t \simeq 300 \cdot 10^6\ Pa$, in agreement with the definitions in Equations(A.15).

After the damage initiation, the effective and nominal stresses σ_t^{eff} and σ_t^{nom}, respectively, follow two different evolutions, in accordance with the definition:

$$\sigma_t^{\text{eff}} = \frac{\sigma_t^{\text{nom}}}{1-d}. \tag{A.20}$$

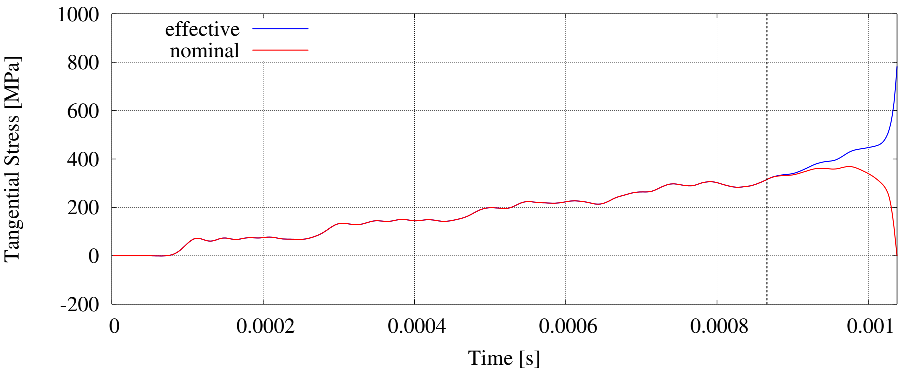


Figure A.4: Tangential stress history at the location S.

Appendix B

Macro-scale model for the Matlab prototype

This appendix describes the formulation and some results of the Enhanced Assumed Strain method for 2D plane strain applications. Such developments have been used for the Matlab prototypes.

Contents

1 Bilinear displacement-based quadrilateral elements

The weak formulation in space of a linear elastic problem with small displacements assumption writes:

$$\int_{\Omega} (\rho \ddot{\mathbf{u}} \cdot \delta \mathbf{u} + \boldsymbol{\sigma} : \delta \boldsymbol{\varepsilon}) \, d\Omega = \int_{\Omega} \mathbf{f} \cdot \delta \mathbf{u} \, d\Omega \tag{B.1}$$

where $\boldsymbol{\sigma}$ denotes the stress tensor defined as:

$$\boldsymbol{\sigma} = \mathbb{D} : \boldsymbol{\varepsilon} \tag{B.2}$$

with the fourth-order tensor $\mathbb{D}$ denoting the elastic constitutive law and with the strain field $\boldsymbol{\varepsilon}$ defined as the symmetric part of the displacement gradient:

$$\boldsymbol{\varepsilon} = \text{grad}^s(\mathbf{u}) = \frac{1}{2}\left(\text{grad}(\mathbf{u}) + \text{grad}^T(\mathbf{u})\right). \tag{B.3}$$

Using a bilinear displacement-based quadrilateral element discretization, the displacement and strain fields in the matrix formulation write:

$$\mathbf{u} = \boldsymbol{\phi} \cdot \mathbf{U} = \mathsf{A}\{\mathbf{N}_{\text{el}} \mathbf{U}_{\text{el}}\}, \qquad \boldsymbol{\varepsilon} = \text{grad}^s(\boldsymbol{\phi}) \cdot \mathbf{U} = \mathsf{A}\{\mathbf{B}_{\text{el}} \mathbf{U}_{\text{el}}\} \tag{B.4}$$

where $\mathsf{A}\{\square\}$ is the assembly operator from elements e to the overall mesh Ω^h, $\mathbf{N}_{\text{el}}$ and $\mathbf{B}_{\text{el}}$ are matrices containing the bilinear shape functions and the related spatial derivatives, respectively.

The mass matrix, stiffness matrix and external forces vector are then respectively defined as:

$$\mathbb{M} = \mathsf{A}\left\{\int_{\Omega_{\text{el}}} \rho \mathbf{N}_{\text{el}}^T \mathbf{N}_{\text{el}} \, d\Omega\right\}, \quad \mathbb{K} = \mathsf{A}\left\{\int_{\Omega_{\text{el}}} \mathbf{B}_{\text{el}}^T \mathbb{D} \mathbf{B}_{\text{el}} \, d\Omega\right\}, \quad \mathbf{F}^{\text{ext}} = \mathsf{A}\left\{\int_{\Omega_{\text{el}}} \mathbf{N}_{\text{el}}^T \mathbf{f} \, d\Omega\right\} \tag{B.5}$$

and Equation (B.1) becomes:

$$\delta \mathbf{U}^T \mathbb{M} \ddot{\mathbf{U}} + \delta \mathbf{U}^T \mathbb{K} \mathbf{U} = \delta \mathbf{U}^T \mathbf{F}^{\text{ext}}. \tag{B.6}$$

With such bilinear shape functions, a parasitic shear strain is numerically produced in bending problems. This kind of error overvalues the stiffness of the structure as described in Figure B.1, with the so-called shear locking effect.

In relation to the theoretical bending behavior described in Figure B.2, the strain solution in the quadrilateral partition reads:

$$\varepsilon_x = -\frac{\vartheta y}{L}, \qquad \varepsilon_y = \frac{\nu \vartheta y}{L}, \qquad \gamma_{xy} = 0. \tag{B.7}$$

On the other hand, with displacement-based quadrilateral elements, the behavior in bending problems is described in Figure B.3 and the strain solution in the quadrilateral element writes:

$$\varepsilon_x = -\frac{\vartheta y}{L}, \qquad \varepsilon_y = 0, \qquad \gamma_{xy} = -\frac{\vartheta x}{L}. \tag{B.8}$$

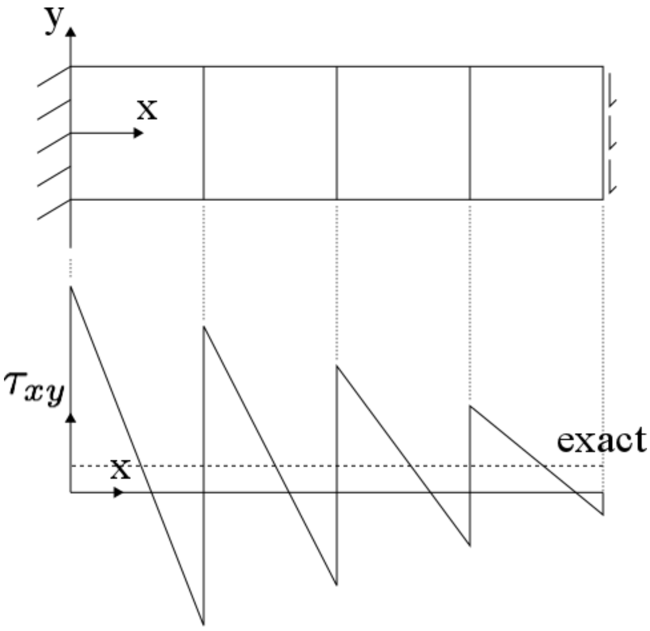


Figure B.1: Bending response in terms of shear stress with the bilinear quadrilateral elements.

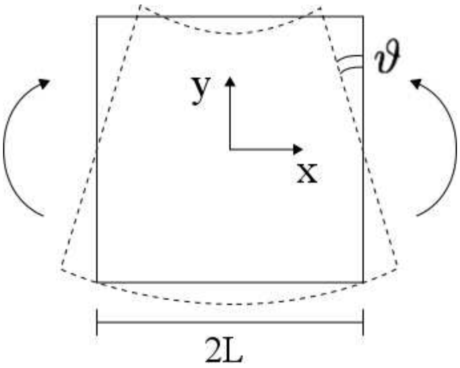


Figure B.2: Theoretical bending behavior neglecting the shear strain.

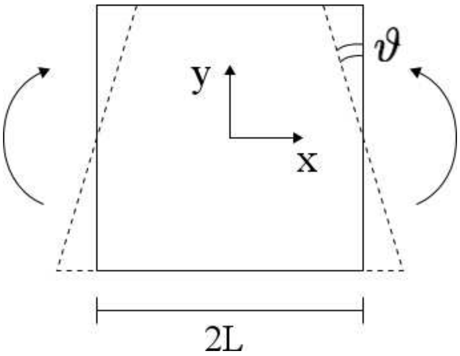


Figure B.3: Bilinear quadrilateral element bending behavior.

2 Enhanced Assumed Strain method

The *Enhanced Assumed Strain* method was formulated in [Simo and Rifai, 1990], with common bases with the study of the incompatible modes in [Taylor et al., 1976] and then enriched of numerical options in [Andelfinger and Ramm, 1993]. An artificial correction is applied to the bilinear quadrilateral element in order to correct the strain and stress responses for bending problems.

The fundamental assumption is that the strain field is decomposed in compatible strains ε, which correspond to the displacements of the Finite Elements space, and an enhancement term $\tilde{\varepsilon}$, so that:

$$\hat{\varepsilon} = \varepsilon + \tilde{\varepsilon} \tag{B.9}$$

that, using the definition in Equation (B.3), becomes:

$$\hat{\varepsilon} = \mathrm{grad}^s(\mathbf{u}) + \tilde{\varepsilon}. \tag{B.10}$$

Focusing the attention on the internal energy term only, the *Hu-Washizu* functional of the linear elastic problem with small displacements assumption writes:

$$\int_\Omega \left(\frac{1}{2}\hat{\varepsilon} : \mathbb{D} : \hat{\varepsilon} - \boldsymbol{\sigma} : \hat{\varepsilon} + \boldsymbol{\sigma} : \varepsilon \right) d\Omega = 0 \tag{B.11}$$

where $\boldsymbol{\sigma}$ is the statically admissible stress field.

Considering the definition in Equation (B.9), Equation (B.11) becomes:

$$\int_\Omega \left(\frac{1}{2}\hat{\varepsilon} : \mathbb{D} : \hat{\varepsilon} - \boldsymbol{\sigma} : \tilde{\varepsilon} \right) d\Omega = 0. \tag{B.12}$$

The crucial assumption in [Simo and Rifai, 1990] is that the additional strains $\tilde{\varepsilon}$ do not contribute to the energy with statically admissible stresses $\boldsymbol{\sigma}$, so that:

$$\int_\Omega \boldsymbol{\sigma} : \tilde{\varepsilon} \, d\Omega = 0. \tag{B.13}$$

Equation (B.12) is so reduced to:

$$\int_\Omega \frac{1}{2}\hat{\varepsilon} : \mathbb{D} : \hat{\varepsilon} \, d\Omega = 0. \tag{B.14}$$

The only variables to be assumed explicitly in Equation (B.14) are the displacements $\mathbf{u}$ and the additional strains $\tilde{\varepsilon}$, according to the definition in Equation (B.10). Whereas the displacements are modeled with bilinear (or trilinear) shape functions as in Equation (B.4), many models are presented and compared for the enhancement term $\tilde{\mathrm{\varepsilon}}$ in [Andelfinger and Ramm, 1993] for 2- and 3-dimensional examples. Using a common formulation, for each element e of the mesh Ω^h:

$$\tilde{\varepsilon}_{\mathrm{el}} = \tilde{\mathbf{M}}_{\mathrm{el}} \boldsymbol{\alpha}_{\mathrm{el}} \tag{B.15}$$

where $\tilde{\mathbf{M}}_{\mathrm{el}}$ is the interpolation matrix for the additional strains and $\boldsymbol{\alpha}_{\mathrm{el}}$ is a vector of internal strain parameters.

In the variational expression in function of parameters $\mathbf{U}$ and $\boldsymbol{\alpha}$, Equation (B.14) for each element e becomes:

$$\begin{aligned}\mathbf{U}_{\mathrm{el}}^{T}\int_{\Omega_{\mathrm{el}}}\mathbf{B}_{\mathrm{el}}^{T}\mathbb{D}\mathbf{B}_{\mathrm{el}}\,d\Omega\;\delta\mathbf{U}_{\mathrm{el}}+\delta\mathbf{U}_{\mathrm{el}}^{T}\int_{\Omega_{\mathrm{el}}}\mathbf{B}_{\mathrm{el}}^{T}\mathbb{D}\tilde{\mathbf{M}}_{\mathrm{el}}\,d\Omega\;\boldsymbol{\alpha}_{\mathrm{el}}+\mathbf{U}_{\mathrm{el}}^{T}\int_{\Omega_{\mathrm{el}}}\mathbf{B}_{\mathrm{el}}^{T}\mathbb{D}\tilde{\mathbf{M}}_{\mathrm{el}}\,d\Omega\;\delta\boldsymbol{\alpha}_{\mathrm{el}}+\\ +\boldsymbol{\alpha}_{\mathrm{el}}^{T}\int_{\Omega_{\mathrm{el}}}\tilde{\mathbf{M}}_{\mathrm{el}}^{T}\mathbb{D}\tilde{\mathbf{M}}_{\mathrm{el}}\,d\Omega\;\delta\boldsymbol{\alpha}_{\mathrm{el}}=0\qquad\forall\delta\mathbf{U}_{\mathrm{el}},\forall\delta\boldsymbol{\alpha}_{\mathrm{el}}\end{aligned}\tag{B.16}$$

that is solved in the two parameter fields $\mathbf{U}_{\mathrm{el}}$ and $\boldsymbol{\alpha}_{\mathrm{el}}$ by:

$$\begin{aligned}\int_{\Omega_{\mathrm{el}}}\mathbf{B}_{\mathrm{el}}^{T}\mathbb{D}\mathbf{B}_{\mathrm{el}}\,d\Omega\;\mathbf{U}_{\mathrm{el}}+\int_{\Omega_{\mathrm{el}}}\mathbf{B}_{\mathrm{el}}^{T}\mathbb{D}\tilde{\mathbf{M}}_{\mathrm{el}}\,d\Omega\;\boldsymbol{\alpha}_{\mathrm{el}}=\mathbf{0}\qquad\forall\delta\mathbf{U}_{\mathrm{el}}\\ \int_{\Omega_{\mathrm{el}}}\tilde{\mathbf{M}}_{\mathrm{el}}^{T}\mathbb{D}\mathbf{B}_{\mathrm{el}}\,d\Omega\;\mathbf{U}_{\mathrm{el}}+\int_{\Omega_{\mathrm{el}}}\tilde{\mathbf{M}}_{\mathrm{el}}^{T}\mathbb{D}\tilde{\mathbf{M}}_{\mathrm{el}}\,d\Omega\;\boldsymbol{\alpha}_{\mathrm{el}}=\mathbf{0}\qquad\forall\delta\boldsymbol{\alpha}_{\mathrm{el}}.\end{aligned}\tag{B.17}$$

Integrating also the inertial energy and the external work at the element level, System (B.17) becomes:

$$\begin{aligned}\mathbb{M}_{\mathrm{el}}\ddot{\mathbf{U}}_{\mathrm{el}}+\mathbb{K}_{\mathrm{el}}\mathbf{U}_{\mathrm{el}}+\mathbb{L}_{\mathrm{el}}\boldsymbol{\alpha}_{\mathrm{el}}=\mathbf{F}_{\mathrm{el}}^{\mathrm{ext}}\\ \mathbb{L}_{\mathrm{el}}^{T}\mathbf{U}_{\mathrm{el}}+\mathbb{Q}_{\mathrm{el}}\boldsymbol{\alpha}_{\mathrm{el}}=\mathbf{0}\end{aligned}\tag{B.18}$$

where:

$$\mathbb{L}_{\mathrm{el}}=\int_{\Omega_{\mathrm{el}}}\mathbf{B}_{\mathrm{el}}^{T}\mathbb{D}\tilde{\mathbf{M}}_{\mathrm{el}}\,d\Omega,\quad\mathbb{Q}_{\mathrm{el}}=\int_{\Omega_{\mathrm{el}}}\tilde{\mathbf{M}}_{\mathrm{el}}^{T}\mathbb{D}\tilde{\mathbf{M}}_{\mathrm{el}}\,d\Omega.\tag{B.19}$$

Condensing the computation of parameters $\boldsymbol{\alpha}_{\mathrm{el}}$, the element problem becomes:

$$\begin{aligned}\boldsymbol{\alpha}_{\mathrm{el}}=-\mathbb{Q}_{\mathrm{el}}^{-1}\mathbb{L}_{\mathrm{el}}^{T}\mathbf{U}_{\mathrm{el}}\\ \left(\mathbb{K}_{\mathrm{el}}-\mathbb{L}_{\mathrm{el}}\mathbb{Q}_{\mathrm{el}}^{-1}\mathbb{L}_{\mathrm{el}}^{T}\right)\mathbf{U}_{\mathrm{el}}=\mathbf{F}_{\mathrm{el}}^{\mathrm{ext}}\end{aligned}\tag{B.20}$$

and the stiffness matrix $\mathbb{K}$ is corrected and assembled as:

$$\mathbb{K}=\mathsf{A}\left\{\mathbb{K}_{\mathrm{el}}-\mathbb{L}_{\mathrm{el}}\mathbb{Q}_{\mathrm{el}}^{-1}\mathbb{L}_{\mathrm{el}}^{T}\right\}.\tag{B.21}$$

According to [Andelfinger and Ramm, 1993], the interpolation matrix $\tilde{\mathbf{M}}_{\mathrm{el}}$ can assume different shapes, in function of the number of parameters $\boldsymbol{\alpha}_{\mathrm{el}}$ chosen for the strain enhancement term $\tilde{\varepsilon}$. For the 2-dimensional case:

1. *EAS-2* with two parameters:

$$\tilde{\mathbf{M}}_{\mathrm{el}}=\begin{bmatrix}\xi&0\\0&\eta\\0&0\end{bmatrix};\tag{B.22}$$

2. *EAS-4 bubble* [Taylor et al., 1976] with four parameters:

$$\tilde{\mathbf{M}}_{\text{el}} = \begin{bmatrix} \xi & 0 & 0 & 0 \\ 0 & \eta & 0 & 0 \\ 0 & 0 & \xi & \eta \end{bmatrix}; \tag{B.23}$$

3. *EAS-5* (with a good improvement of bending behavior) with five parameters:

$$\tilde{\mathbf{M}}_{\text{el}} = \begin{bmatrix} \xi & 0 & 0 & 0 & \xi\eta \\ 0 & \eta & 0 & 0 & -\xi\eta \\ 0 & 0 & \xi & \eta & \xi^2+\eta^2 \end{bmatrix}; \tag{B.24}$$

4. *EAS-7* [Andelfinger and Ramm, 1993] with seven parameters:

$$\tilde{\mathbf{M}}_{\text{el}} = \begin{bmatrix} \xi & 0 & 0 & 0 & \xi\eta & 0 & 0 \\ 0 & \eta & 0 & 0 & 0 & \xi\eta & 0 \\ 0 & 0 & \xi & \eta & 0 & 0 & \xi\eta \end{bmatrix}. \tag{B.25}$$

3 Study of the convergence with mesh refinement

The problem displayed in Figure B.4 is used to test the accuracy of the *EAS* quadrilateral elements.

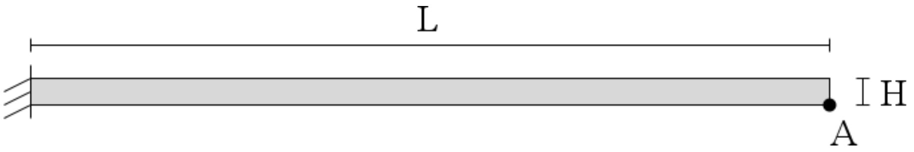


Figure B.4: Thin cantilever beam subjected to body forces.

A cantilever beam with plane strain assumption subjected to body forces is analyzed with different spatial discretizations varying the number of elements throughout the length L of the domain, value that influences the shape of the elements.

The problem data used in the simulations are:

- density $\rho = 7500\ kg/m^3$;
- Young's modulus $E = 210 \cdot 10^9\ Pa$;
- Poisson's ratio $\nu = 0.3$;
- length $L = 50\ m$;
- height $H = 0.5\ m$;
- initial time $t_{\text{in}} = 0\ s$;

- final time $t_{\text{end}} = 10\ s$;
- body force $g = 9.81\ m/s^2$.

Denoting their aspect-ratio as $a_r = L_e/H_e$:

- 1 element through the length $\Rightarrow\ a_r = 100$;
- 2 elements through the length $\Rightarrow\ a_r = 50$;
- 4 elements through the length $\Rightarrow\ a_r = 25$;
- 10 elements through the length $\Rightarrow\ a_r = 10$;
- 100 elements through the length $\Rightarrow\ a_r = 1$.

Remark 10 *When the aspect-ratio a_r is high, the application of the Selective Mass Scaling technique introduced in [Cocchetti et al., 2012] for the case of 3-dimensional solid-like shell formulations is desired, in order to increase the critical time step linked to the small thickness. Such technique is based on the linear transformation of the element degrees of freedom, for allowing an artificial mass scaling technique to selectively reduce the highest structural eigenfrequencies.*

A time step close to the critical one is fixed as $\Delta t = 5 \cdot 10^{-5}\ s$ for all the simulations.

Figure B.5 shows the convergence of the solutions refining the mesh, measured taking the *EAS-7* quadrilateral element with aspect-ratio $a_r = 1$ as reference. The curves show a remarkable enhancement of the accuracy with the *EAS-4 bubble*, *EAS-5* and *EAS-7* elements in comparison with the *EAS-2* and the classical displacement-based elements. The *EAS-4 bubble* results the most convenient element technology, because requires less parameters for a similar accuracy to *EAS-5* and *EAS-7*.

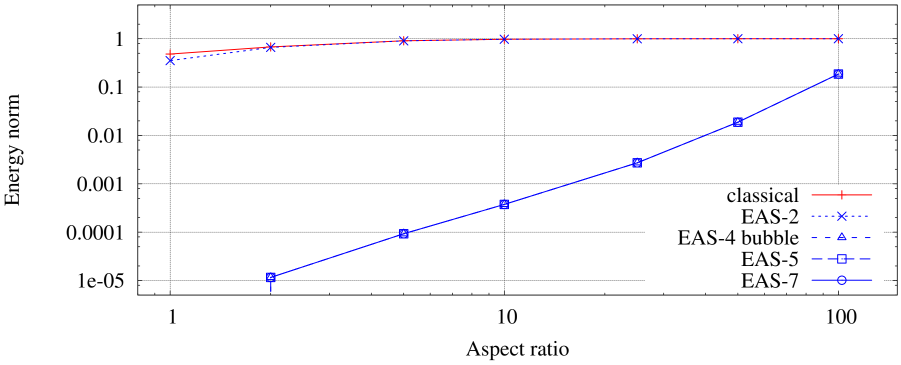
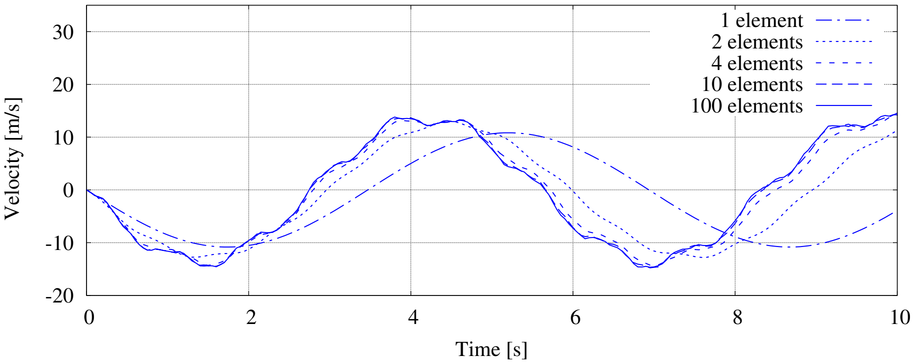
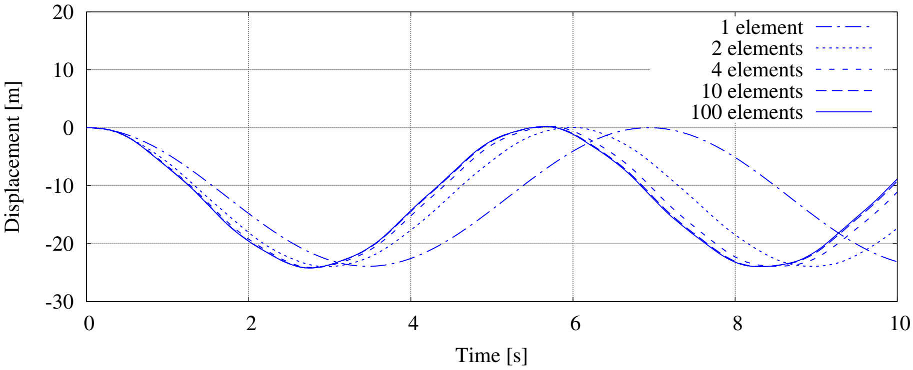


Figure B.5: Convergence refining the mesh.

Results in terms of velocities and displacements are shown in Figure B.6 and denote a fast convergence to the exact solution refining the spatial discretization, so that the meshes with 10 elements ($a_r = 10$) or 100 elements ($a_r = 1$) produce almost the same solution in terms of vertical displacements.



(a) vertical velocities



(b) vertical displacements

Figure B.6: Solution at the location A.

Appendix C

Implementation inside Abaqus

This appendix describes the implementation of the Substitution method inside the commercial software Abaqus, implemented in order to test the compatibility of the coupling technique with the Abaqus code architecture. The application to truss elements is proposed.

Contents

1 Preliminary discussions and tests

The compatibility of the Substitution method with the existing features has been preliminarily discussed, meaning that fundamental is the possibility to deal with:

- geometric or constitutive nonlinearities and artificial viscous forces for high frequency dissipation: the iterative scheme is not affected because it only influences the definition of the nodal forces applied to the interface nodes;
- non-conforming meshes with misaligned interfaces to the global elements boundaries: the insertion of slave nodes in the global mesh corresponding to the interface and the numerical integration of the partitions of the element so created can be employed to define the required nodal forces; further investigations could verify the feasibility of such implementation or the definition of an equivalent less intrusive strategy;
- contact problems: the contact is considered only in the global analysis and the equivalent global forces are mapped onto the local model; this approach is convenient because it allows one to model the impactor and the contact process with the coarse model: further investigation could interest the treatment of the contact at the local level in order to increase the accuracy of the results.

2 Co-Simulation technique for Domain Decomposition implicit-explicit coupling

As introduced in Section 5.4 of Chapter 1, the Co-Simulation can be utilized for Domain Decomposition implicit-explicit couplings via the *GC* algorithm. Each subdomain is run with its own analysis package. In the simplest case with two subdomains, the coarse one is run with Abaqus/Standard, whereas the finer one with Abaqus/Explicit. The necessary functionalities are in the Abaqus/Standard environment and include:

- .cpp services for computations and variables manipulations;
- .h objects for variables definition;
- .xml *state chart* [W3C, 2014] for governing the algorithm flow chart and the data exchanges.

3 Extension to Domain Decomposition explicit-explicit coupling

At first, the Co-Simulation technique for Domain Decomposition has been extended to explicit-explicit couplings. Such extension has required the implementation of the functionalities in the Abaqus/Explicit environment, in which variables and operations are deeply dissimilar.

The multi-time-step FETI approach formulated in Section 2 of Chapter 2 has been employed.

The main steps of the explicit-explicit coupling via the Co-Simulation are summarized in Algorithm 13.

```
while $n < N$ do
    coarse subdomain: solution of free momentum balance;
    coarse subdomain sends to fine subdomain: mass and velocity;
    while $m < M$ do
        fine subdomain: solution of free momentum balance;
        service: computation of Lagrange multipliers;
        fine subdomain: link correction with Lagrange multipliers;
        update $m$;
    end
    fine subdomain sends to coarse subdomain: Lagrange multipliers;
    coarse subdomain: link correction with Lagrange multipliers;
    update $n$;
end
```

Algorithm 13: Co-Simulation for /Explicit-/Explicit Domain Decomposition.

4 Substitution via the Co-Simulation technique

The implementation of the Substitution method via the Co-Simulation technique requires the same type of functionalities of the Domain Decomposition explicit-explicit coupling, even if the computations, the data exchange and the flow chart are strongly different.

The main steps of the explicit-explicit coupling are summarized in Algorithm 14.

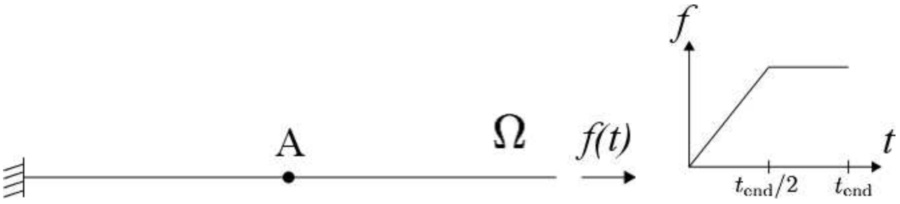
```
while $n < N$ do
    global analysis: solution of momentum balance as pre-computation;
    while $i < \bar{i}$ do
        global sends to local: acceleration and velocity;
        back in time;
        while $m < M$ do
            local analysis: solution of momentum balance;
            service: imposition of global solution over the interface;
            update $m$;
        end
        local sends to global: reaction over the interface;
        service: correction of the global solution over the interface;
        service: direct substitution in the global substitution region;
    end
    update $n$;
end
```

Algorithm 14: Co-Simulation for /Explicit-/Explicit Substitution.

Note that, for the sake of simplicity, a fixed number of iterations $\bar{i}$ is considered, so to avoid the computation of the convergence measure e.

5 Application to truss elements

The domain problem illustrated in Figure C.1 is considered as preliminary test of the Substitution method inside Abaqus.



(a) spatial domain (b) load in time

Figure C.1: Truss test example: boundary conditions, load and monitored location A.

A bar with uniform cross section is clamped at the left side and pulled at the right side by a force variable in time. The following data are employed:

- density $\rho = 7500\ kg/m^3$;
- Young's modulus $E = 210 \cdot 10^9\ Pa$;
- cross section $A = 2\ m^2$;

- length $L = 10\ m$;
- initial time $t_{\text{in}} = 0\ s$;
- final time $t_{\text{end}} = 0.01\ s$;
- maximum load $\overline{f} = 10^9\ N$.

A monolithic analysis is performed for reference results with the non-uniform mesh Ω_r^h displayed in Figure C.2. One truss element covers the left side of the bar, whereas five truss elements of same length are applied to the right side. A unique time step $\Delta t_r = 1 \cdot 10^{-4}\ s$ is fixed to in the overall domain for all the duration of the analysis.

Figure C.2: Monolithic analysis with non-uniform mesh and uniform time step.

The Domain Decomposition analysis via the Co-Simulation technique is then performed with the two subdomains displayed in Figure C.3. One truss element is used in the first subdomain Ω_1^h at the left side of the bar, whereas five truss elements of same length are applied to the second subdomain Ω_2^h on the right side. Two different time steps are defined to the two subdomains, such that:

- $\Delta t_1 = 5 \cdot 10^{-4}\ s$ in the coarse subdomain Ω_1^h;
- $\Delta t_2 = 1 \cdot 10^{-4}\ s$ in the fine subdomain Ω_2^h.

The exchanges between the two subdomains are represented by the green bi-directional arrow, indicating also the interface nodes of the two subdomains.

Figure C.3: Meshes for the Domain Decomposition.

The Substitution analysis via the Co-Simulation technique is finally performed with the global and local meshes displayed in Figure C.4. The global analysis at the top is modeled by a uniform global mesh Ω_g^h of two truss elements, the second one of them is overlapped by the uniform local mesh Ω_ℓ^h at the bottom constituted by five truss elements. Similarly to the Domain Decomposition approach, two time steps are used in the two analyses:

- $\Delta t_g = 5 \cdot 10^{-4}\ s$ in the coarse subdomain Ω_g^h;
- $\Delta t_\ell = 1 \cdot 10^{-4}\ s$ in the fine subdomain Ω_ℓ^h.

The exchanges between the two analyses are represented by the red bi-directional arrow for the *iterative substitution* algorithm and the blue arrow for the *direct substitution* algorithm.

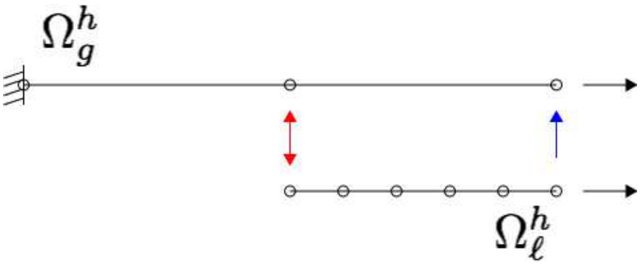
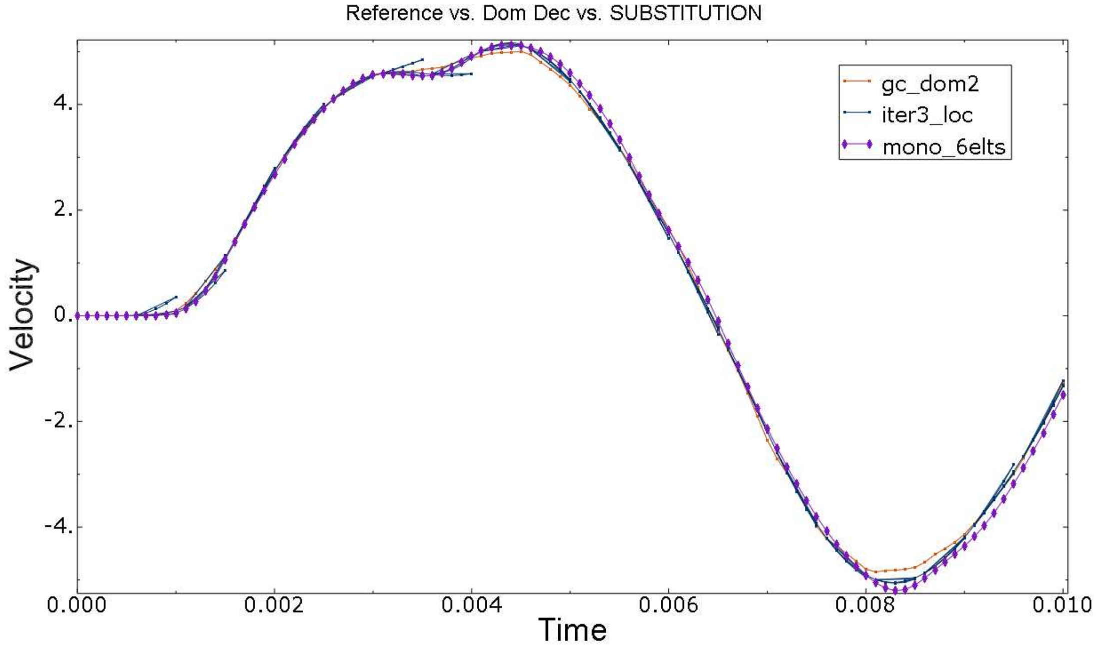
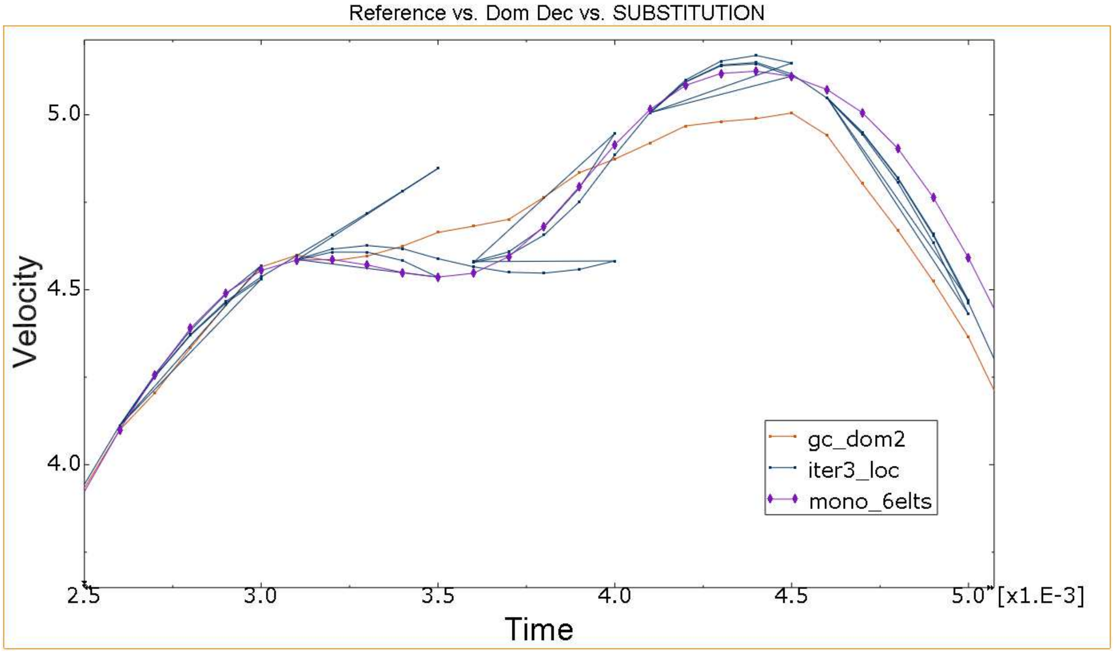


Figure C.4: Meshes for the Substitution.

Figures C.5 show the comparison between the three above mentioned analyses in terms of velocity results at the location A, that is also the interface position for the Domain Decomposition and the Substitution methods. All the results are taken in the fine time scale. The purple lines “mono_6elts” denote the monolithic analysis results with uniform time step and are taken as reference curves. The orange lines “gc_dom2” denote the Domain Decomposition analysis results. The blue lines “iter3_loc” denote the Substitution analysis results, in which the a pre-fixed number of iterations $\bar{i} = 3$ has been imposed. Every iteration, from the first one to the third, is plotted. One can spot the improved accuracy of the iterative scheme after the first iteration and after the second as well: the path that has been chosen to be accurate enough is represented by the outputs of the third iterations. In the plot at the top, which show the whole analysis, the curves approximately follow the same path. In the plot at the bottom, expanding the results between the times $t_1 = 2.5 \cdot 10^{-3}\ s$ and $t_2 = 5.0 \cdot 10^{-3}\ s$ around the stationary points of the velocity results, the Substitution curve is closer to the reference solution than the Domain Decomposition one, because of the more accurate constraint at the interface. The gap between Substitution and reference analyses is due to the time step discrepancies in the coarse zones, that is finer in the reference outcome.



(a) entire analysis



(b) expansion around the stationary points between $t_1 = 2.5 \cdot 10^{-3}\ s$ and $t_2 = 5 \cdot 10^{-3}\ s$

Figure C.5: Velocities at the location A.

Bibliography

[Allix, 2001] Allix, O. (2001). A composite damage meso-model for impact problems. *Composites Science and Technology*, 61:2193–2205.

[Allix and Blanchard, 2006] Allix, O. and Blanchard, L. (2006). Mesomodeling of delamination: towards industrial applications. *Composites Science and Technology*, 66(6):731–744.

[Allix and Ladevèze, 1992] Allix, O. and Ladevèze, P. (1992). Interlaminar interface modelling for the prediction of delamination. *Composite Structures*, 22:235–242.

[Allix et al., 1995] Allix, O., Ladevèze, P., and Corigliano, A. (1995). Damage analysis of interlaminar fracture specimens. *Composite Structures*, 31:61–74.

[Andelfinger and Ramm, 1993] Andelfinger, U. and Ramm, E. (1993). EAS-elements for two dimensional, three-dimensional, plate and shell structures and their equivalence to HR-elements. *International Journal for Numerical Methods in Engineering*, 36:1311–1337.

[Armero and Linder, 2009] Armero, F. and Linder, C. (2009). Numerical simulation of dynamic fracture using finite elements with embedded discontinuities. *International Journal of Fracture*, 160(2):119–141.

[Bathe, 1996] Bathe, K. J. (1996). *Finite Element Procedures*. Prentice-Hall, Inc.

[Bathe and Wilson, 1973] Bathe, K. J. and Wilson, E. L. (1973). Stability and accuracy analysis of direct integration methods. *Earthquake Engineering and Structural Dynamics*, 1:283–291.

[Belytschko and Mullen, 1978] Belytschko, T. and Mullen, R. (1978). Stability of explicit-implicit mesh partitions in time integration. *International Journal for Numerical Methods in Engineering*, 12:15751586.

[Belytschko et al., 1985] Belytschko, T., Smolinski, P., and Liu, W. K. (1985). Stability of multi-time step partitioned integrators for first-order finite element systems. *Computer Methods in Applied Mechanics and Engineering*, 49:281–297.

[Belytschko et al., 1979] Belytschko, T., Yen, H.-J., and Mullen, R. (1979). Mixed methods for time integration. *Computer Methods in Applied Mechanics and Engineering*, 17-18:259–275.

[BenDhia and Rateau, 2005] BenDhia, H. and Rateau, G. (2005). The Arlequin method as a flexible engineering design tool. *International Journal for Numerical Methods in Engineering*, 62:1442–1462.

[Bettinotti et al., 2014a] Bettinotti, O., Allix, O., and Malherbe, B. (2014a). A coupling strategy for adaptive local refinement in space and time with a fixed global model in explicit dynamics. *Computational Mechanics*, 53:561–574.

[Bettinotti et al., 2014b] Bettinotti, O., Allix, O., Perego, U., Oancea, V., and Malherbe, B. (2014b). A fast weakly intrusive multiscale method in explicit dynamics. *International Journal for Numerical Methods in Engineering, accepted.*

[Biabanaki et al., 2014] Biabanaki, S., Khoei, A., and Wriggers, P. (2014). Polygonal finite element methods for contact-impact problems on non-conformal meshes. *Computer Methods in Applied Mechanics and Engineering*, 269:198 – 221.

[Boucinha et al., 2014] Boucinha, L., Ammar, A., Gravouil, A., and Nouy, A. (2014). Ideal minimal residual-based proper generalized decomposition for non-symmetric multi-field models - application to transient elastodynamics in space-time domain. *Computational Methods in Applied Mechanics and Engineering*, 273:56–76.

[Boucinha et al., 2013] Boucinha, L., Gravouil, A., and Ammar, A. (2013). Space-time proper generalized decompositions for the resolution of transient elastodynamic models. *Computational Methods in Applied Mechanics and Engineering*, 255:67–88.

[Brandt, 1977] Brandt, A. (1977). Multi-level adaptive solutions to boundary-value problems. *Mathematics of Computation*, 31(138):333–390.

[Campello et al., 2011] Campello, E. M. B., Pimenta, P. M., and Wriggers, P. (2011). An exact conserving algorithm for nonlinear dynamics with rotational DOFs and general hyperelasticity. Part 2: shells. *Computational Mechanics*, 48(2):195–211.

[Champaney et al., 1997] Champaney, L., Cognard, J. Y., Dureisseix, D., and Ladevèze, P. (1997). Large scale applications on parallel computers of a mixed domain decomposition method. *Computational Mechanics*, 19(4):253–263.

[Chantrait et al., 2014] Chantrait, T., Rannou, J., and Gravouil, A. (2014). Low intrusive coupling of implicit and explicit time integration schemes for structural dynamics: Application to low energy impacts on composite structures. *Finite Elements in Analysis and Design, to appear*, 86:23–33.

[Chung and Lee, 1994] Chung, J. and Lee, J. M. (1994). A new family of explicit time integration methods for linear and non-linear structural dynamics. *International Journal for Numerical Methods in Engineering*, 37(23):3961–3976.

[Cocchetti et al., 2012] Cocchetti, G., Pagani, M., and Perego, U. (2012). Selective mass scaling and critical time-step estimate for explicit dynamics analyses with solid-shell elements. *Computers and Structures*, 127:39–52.

[Combescure and Gravouil, 2001] Combescure, A. and Gravouil, A. (2001). A time-space multi-scale algorithm for transient structural nonlinear problems. *Editions scientifiques et medicales*, 2(1):43–55.

[Combescure and Gravouil, 2002] Combescure, A. and Gravouil, A. (2002). A numerical scheme to couple subdomains with different time-steps for predominantely transient linear analysis. *Computer Methods in Applied Mechanics and Engineering*, 191(11-12):1129–1157.

[Courant et al., 1967] Courant, R., Friedrichs, K., and Lewy, H. (1967). On the partial difference equations of mathematical physics. *IBM Journal of Research and Development*, 11(2):215–234.

[Cresta et al., 2007] Cresta, P., Allix, O., Rey, C., and Guinard, S. (2007). Nonlinear localization strategies for domain decomposition methods: Application to post-buckling analyses. *Computer Methods in Applied Mechanics and Engineering*, 196:1436–1446.

[Daniel, 1997] Daniel, W. J. T. (1997). Analysis and implementation of a new constant acceleration subcycling algorithm. *International Journal for Numerical Methods in Engineering*, 40(15):2841–2855.

[Daniel, 2003] Daniel, W. J. T. (2003). A partial velocity approach to subcycling structural dynamics. *Computer Methods in Applied Mechanics and Engineering*, 192(3):375–394.

[Dureisseix and Ladevèze, 1998] Dureisseix, D. and Ladevèze, P. (1998). A 2-level and mixed domain decomposition approach for structural analysis. *Contemporary Mathematics*, 218:238–244.

[Farhat et al., 1994] Farhat, C., Crivelli, L., and Roux, F.-X. (1994). A transient FETI methodology for large-scale parallel implicit computations in structural mechanics. *International Journal for Numerical Methods in Engineering*, 37(11):1945–1975.

[Farhat and Roux, 1991] Farhat, C. and Roux, F.-X. (1991). A method of finite element tearing and interconnecting and its parallel solution algorithm. *International Journal for Numerical Methods in Engineering*, 32:1205–1227.

[Fish and Belsky, 1995] Fish, J. and Belsky, V. (1995). Multigrid method for periodic heterogeneous media. *Computer Methods in Applied Mechanics and Engineering*, 126:1–38.

[Fish et al., 1996] Fish, J., Belsky, V., and Pandheeradi, M. (1996). Composite grid method for hybrid systems. *Computer Methods in Applied Mechanics and Engineering*, 135:307–325.

[Fish et al., 1997] Fish, J., Suvorov, A., and Belsky, V. (1997). Hierarchical composite grid method for global-local analysis of laminated composite shells. *Applied Numerical Mathematics*, 23:241–258.

[Flanagan and Belytschko, 1985] Flanagan, D. P. and Belytschko, T. (1985). Eigenvalues and stable time steps for the uniform strain hexahedron and quadrilateral. *Journal of Applied Mechanics*, 51(1):35–40.

[Gendre et al., 2011] Gendre, L., Allix, O., and Gosselet, P. (2011). A two-scale approximation of the Schur complement and its use for non-intrusive coupling. *International Journal for Numerical Methods in Engineering*, 87:889–905.

[Gendre et al., 2009] Gendre, L., Allix, O., Gosselet, P., and Comte, F. (2009). Non-intrusive and exact global/local techniques for structural problems with local plasticity. *Computational Mechanics*, pages 233–245.

[Ghanem et al., 2013] Ghanem, A., Torkhani, M., Mahjoubi, N., Baranger, T. N., and Combescure, A. (2013). Arlequin framework for multi-model, multi-time scale and heterogeneous time integrators for structural transient dynamics. *Computational Methods in Applied Mechanics and Engineering*, 254:292–308.

[Glowinski et al., 2005] Glowinski, R., He, J., Lozinski, A., Rappaz, J., and Wagner, J. (2005). Finite element approximation of multi-scale elliptic problems using patches of elements. *Numerische Mathematik*, 101:663–687.

[Gosselet and Rey, 2006] Gosselet, P. and Rey, C. (2006). Non-overlapping domain decomposition methods in structural mechanics. *Archives of Computational methods in Engineering*, 13(5):515–572.

[Gosselet et al., 2003] Gosselet, P., Rey, C., and Rixen, D. (2003). On the initial estimate of interface forces in FETI methods. *Computer Methods in Applied Mechanics and Engineering*, 192:2749–2764.

[Gravouil and Combescure, 2001] Gravouil, A. and Combescure, A. (2001). Multi-time-step explicit-implicit method for non-linear structural dynamics. *International Journal for Numerical Methods in Engineering*, 50:199–225.

[Guguin et al., 2014] Guguin, G., Allix, O., Gosselet, P., and Guinard, S. (2014). Non-intrusive coupling of 3D and 2D laminated composite models based on finite element 3d recovery. *International Journal for Numerical Methods in Engineering*, 98(5):324–343.

[Guidault et al., 2007] Guidault, P.-A., Allix, O., Champaney, L., and Navarro, J.-P. (2007). A two-scale approach with homogeneization for the computation of cracked structures. *Computer and Structures*, 85(17-18):1360–1371.

[Guimard et al., 2009] Guimard, J.-M., Allix, O., Pechnik, N., and Thévenet, P. (2009). Characterization and modeling of rate effects in the dynamic propagation of mode-II delamination in composite laminates. *International Journal of Fracture*, 160:55–71.

[Gupta et al., 2012] Gupta, P., Pereira, J. P., Kim, D.-J., Duarte, C. A., and Eason, T. (2012). Analysis of three-dimensional fracture mechanics problems: A non-intrusive approach using a generalized finite element method. *Engineering Fracture Mechanics*, 90:41–64.

[Hallquist, 1998] Hallquist, J. O. (1998). *LS-Dyna Theoretical manual*. Livermore Software Technology Corporation.

[Heimbs, 2011] Heimbs, S. (2011). Fe models and reports concerning impactor database, study on pre-stress effect and multi-scale modeling strategy. part 3: Multi-scale modeling stategies. *MAAXIMUS-WP3.5*, pages 1–58.

[Heimbs et al., 2014] Heimbs, S., Bergmann, T., Schueler, D., and Toso-Pentecôte, N. (2014). High velocity impact on preloaded composite plates. *Composite Structures*, 111:158–168.

[Hibbitt and Karlsson, 1979] Hibbitt, H. D. and Karlsson, B. I. (1979). *Analysis of a Pipe Whip, EPRI Report, NP-1208, Electric Power Research Institute*. The Institute.

[Hilber et al., 1977] Hilber, H. M., Hughes, T. J. R., and Taylor, R. L. (1977). Improved numerical dissipation for time integration algorithms in structural dynamics. *Earthquake Engineering and Structural Dynamics*, 5:282–292.

[Hinojosa et al., 2014] Hinojosa, J., Allix, O., Guidault, P.-A., and Cresta, P. (2014). Domain decomposition methods with nonlinear localization for the buckling and post-buckling analyses of large structures. *Advances in Engineering Software*, 70:13–24.

[Houbolt, 1950] Houbolt, J. C. (1950). A recurrence matrix solution for the dynamic response of elastic aircraft. *Journal of the Aeronautical Sciences*, 17:540–550.

[Hughes, 1987] Hughes, T. J. R. (1987). *The Finite Element Method, linear static and dynamic Finite Element Analysis*. Prentice-Hall, Inc.

[Hughes and Liu, 1978a] Hughes, T. J. R. and Liu, W. K. (1978a). Implicit-explicit finite element in transient analysis: implementation and numerical examples. *Journal of Applied Mechanics*, 45:375–379.

[Hughes and Liu, 1978b] Hughes, T. J. R. and Liu, W. K. (1978b). Implicit-explicit finite element in transient analysis: stability theory. *Journal of Applied Mechanics*, 45:371–374.

[Hulbert and Chung, 1996] Hulbert, G. M. and Chung, J. (1996). Explicit time integration algorithms for structural dynamics with optimal numerical dissipation. *Computer Methods in Applied Mechanics and Engineering*, 137:175–188.

[Hulbert and Hughes, 1990] Hulbert, G. M. and Hughes, T. J. R. (1990). Space-time finite element methods for second-order hyperbolic equations. *Computer Methods in Applied Mechanics and Engineering*, 84:327–348.

[Kerfriden et al., 2009] Kerfriden, P., Allix, O., and Gosselet, P. (2009). A three-scale domain decomposition method for the 3D analysis of debonding in laminates. *Computational Mechanics*, 44(3):343–362.

[Kerfriden et al., 2012] Kerfriden, P., Passieux, J. C., and Bordas, S. P. A. (2012). Local/global model order reduction strategy for the simulation of quasi-brittle fracture. *International Journal for Numerical Methods in Engineering*, 89(2):154–179.

[Ladevèze et al., 2000] Ladevèze, P., Allix, O., Deü, J.-F., and Lévèque, D. (2000). A meso-model for localization and damage computation in laminates. *Computer Methods in Applied Mechanics and Engineering*, 183:105–122.

[Ladevèze and LeDantec, 1992] Ladevèze, P. and LeDantec, E. (1992). Damage modelling of the elementary ply for laminated composites. *Composite Science and Technology*, 43:257–267.

[Ladevèze et al., 2001] Ladevèze, P., Loiseau, O., and Dureisseix, D. (2001). A micro-macro and parallel computational strategy for highly heterogeneous structures. *International Journal for Numerical Methods in Engineering*, 52(1-2):121–138.

[Ladevèze and Nouy, 2003] Ladevèze, P. and Nouy, A. (2003). On a multiscale computational strategy with time and space homogenization for structural mechanics. *Computer Methods in Applied Mechanics and Engineering*, 192:3061–3087.

[Ladevžee, 1985] Ladevžee, P. (1985). On algorithm family in structural mechanics. *Comptes Rendus de l Acadmie des Sciences - Series IIB - Mechanics-Physics-Chemistry-Astronomy*, 300:41–44.

[Landshoff, 1955] Landshoff, R. (1955). A numerical method for treating fluid flow in the presence of shocks. *Technical Report LA-1930, Los Alamos National Laboratory*, pages 1–40.

[Laursen and Meng, 2001] Laursen, T. A. and Meng, X. N. (2001). A new solution procedure for application of energy-conserving algorithms to general constitutive models in nonlinear elastodynamics. *Computer Methods in Applied Mechanics and Engineering*, 190(46-47):6309–6322.

[LeTallec et al., 1991] LeTallec, P., DeRoeck, Y.-H., and Vidrascu, M. (1991). Domain-decomposition method for large linearly elliptic three dimensional problems. *Journal of Computational and Applied Mathematics*, 34:93–117.

[Mahjoubi et al., 2009] Mahjoubi, N., Gravouil, A., and Combescure, A. (2009). Coupling subdomains with heterogeneous time integrators and incompatible time steps. *Computational Mechanics*, 44:825–843.

[Mahjoubi et al., 2011] Mahjoubi, N., Gravouil, A., Combescure, A., and Greffet, N. (2011). A monolithic energy conservating method to couple heterogeneous time integrators with incompatible time steps in structural dynamics. *Computer Methods in Applied Mechanics and Engineering*, 200:1069–1086.

[Mandel, 1993] Mandel, J. (1993). Balancing domain decomposition. *International Journal for Numerical Methods in Biomedical Engineering*, 9(3):233–241.

[McCormick and Ruge, 1982] McCormick, S. and Ruge, J. (1982). Multigrid methods for variational problems. *Journal on Numerical Analysis*, 19(2):924–929.

[Newmark, 1959] Newmark, N. M. (1959). A method of computation for structural dynamics. *Journal of the Engineering Mechanics Division*, 85(7):67–94.

[Noh and Bathe, 2013] Noh, G. and Bathe, K.-J. (2013). An explicit time integration scheme for the analysis of wave propagations. *Computers and Structures*, 129:178–193.

[Nour-Omid and Wriggers, 1986] Nour-Omid, B. and Wriggers, P. (1986). A two-level iteration method for solution of contact problems. *Computer Methods in Applied Mechanics and Engineering*, 54:131–144.

[Oancea and Laursen, 1997] Oancea, V. G. and Laursen, T. A. (1997). A finite element formulation of thermomechanical rate-dependent frictional sliding. *International Journal for Numerical Methods in Engineering*, 40:4275–4311.

[Odièvre et al., 2010] Odièvre, D., Boucard, P.-A., and Gatuingt, F. (2010). A parallel, multi scale domain decomposition method for the transient dynamic analysis of assemblies with friction. *Computer Methods in Applied Mechanics and Engineering*, 199:1297–1306.

[Oliver et al., 2003] Oliver, J., Huespe, A. E., Pulido, M. D. G., and Samaniego, E. (2003). On the strong discontinuity approach in finite deformation settings. *International Journal for Numerical Methods in Engineering*, 56:1051–1082.

[Parsons and Hall, 1990] Parsons, I. D. and Hall, J. F. (1990). The multigrid method in solid mechanics. *International Journal for Numerical Methods in Engineering*, 29(4):719–753.

[Passieux et al., 2013] Passieux, J.-C., Réthoré, J., Gravouil, A., and Baietto, M.-C. (2013). Local/global non-intrusive crack propagation simulation using a multigrid X-FEM solver. *Computational Mechanics*, 52(6):1381–1393.

[Pimenta et al., 2008] Pimenta, P. M., Campello, E. M. B., and Wriggers, P. (2008). An exact conserving algorithm for nonlinear dynamics with rotational DOFs and general hyperelasticiy. Part 1: Rods. *Computational Mechanics*, 42:715–732.

[Plews et al., 2012] Plews, J., Duarte, C. A., and Eason, T. (2012). An improved nonintrusive global-local approach for sharp thermal gradients in a standard FEA platform. *International Journal for Numerical Methods in Engineering*, 91(4):361–397.

[Prakash and Hjelmstad, 2004] Prakash, A. and Hjelmstad, K. D. (2004). A FETI-based multi-time-step coupling method for Newmark schemes in structural dynamics. *International Journal for Numerical Methods in Engineering*, 61:2183–2204.

[Rio et al., 2005] Rio, G., Soive, A., and Grolleau, V. (2005). Comparative study of numerical explicit time integration algorithms. *Advances in Engineering Software*, 36(4):252–265.

[Rixen, 2002] Rixen, D. (2002). Extended preconditioners for the FETI method applied to constrained problems. *International Journal for Numerical Methods in Engineering*, 54(1):1–26.

[Rixen and Farhat, 1999] Rixen, D. and Farhat, C. (1999). A simple and efficient extension of a class of substructure based preconditioners to heterogeneous structural mechanics problems. *International Journal for Numerical Methods in Engineering*, 44(4):489–516.

[Saavedra et al., 2012] Saavedra, K., Allix, O., and Gosselet, P. (2012). On a multiscale strategy and its optimization for the simulation of combined delamination and buckling. *International Journal for Numerical Methods in Engineering*, 91(7):772–798.

[Sansour et al., 2002] Sansour, C., Wagner, W., Wriggers, P., and Sansour, J. (2002). An energymomentum integration scheme and enhanced strain finite elements for the nonlinear dynamics of shells. *International Journal of Non-Linear Mechanics*, 37:951–966.

[Schellekens and de Borst, 1993a] Schellekens, J. C. J. and de Borst, R. (1993a). A nonlinear finite element approach for the analysis of mode-I free edge delamination in composites. *International Journal of Solids and Structures*, 30(9):1239–1253.

[Schellekens and de Borst, 1993b] Schellekens, J. C. J. and de Borst, R. (1993b). On the numerical integration of interface elements. *International Journal for Numerical Methods in Engineering*, 36:43–66.

[Simo and Rifai, 1990] Simo, J. C. and Rifai, M. S. (1990). A class of assumed strain methods an the method of incompatible modes. *International Journal for Numerical Methods in Engineering*, 29:1595–1638.

[Simo and Tarnow, 1992] Simo, J. C. and Tarnow, N. (1992). The discrete energy-momentum method. Conserving algorithms for nonlinear elastodynamics. *Zeitschrift fr angewandte Mathematik und Physik*, 43:757–792.

[Simo and Tarnow, 1994] Simo, J. C. and Tarnow, N. (1994). A new energy and momentum conserving algorithm for the non-linear dynamics of shells. *International Journal for Numerical Methods in Engineering*, 37(15):2527–2549.

[Simo et al., 1995] Simo, J. C., Tarnow, N., and Doblare, M. (1995). Non-linear dynamics of three-dimensional rods: Exact energy and momentum conserving algorithms. *International Journal for Numerical Methods in Engineering*, 38(9):1431–1473.

[Smolinski et al., 1988] Smolinski, P., Belytschko, T., and Neal, M. (1988). Multi-time-step integration using nodal partitioning. *International Journal for Numerical Methods in Engineering*, 26:349–359.

[Smolinski et al., 1996] Smolinski, P., Sleith, S., and Belytschko, T. (1996). Explicit-subcycling for linear structural dynamics. *Computational Mechanics*, 18:236–244.

[Sukumar et al., 2000] Sukumar, N., Moës, N., Moran, B., and Belytschko, T. (2000). Extended Finite Element method for three-dimensional crack modeling. *International Journal for Numerical Methods in Engineering*, 48(11):1549–1570.

[Swope et al., 1982] Swope, W. C., Andersen, H. C., Berens, P. H., and Wilson, K. R. (1982). A computer simulation method for the calculation of equilibrium constants for the formation of physical clusters of molecules: Application to small water clusters. *The Journal of Chemical Physics*, 76(1):637–649.

[Taylor et al., 1976] Taylor, R. L., Beresford, P. J., and Wilson, E. L. (1976). A nonconforming element for stress analysis. *International Journal for Numerical Methods in Engineering*, 10:1211–1219.

[Tchamwa et al., 1999] Tchamwa, B., Conway, T., and Wielgosz, C. (1999). An accurate explicit direct time integration method for computational structural dynamics. *Recent advances in solids and structures*, 398:77–84.

[To and Li, 2005] To, A. C. and Li, S. F. (2005). Perfectly matched multiscale simulations. *Physical Review B*, 72(035414).

[Verlet, 1967] Verlet, L. (1967). Thermodynamical properties of Lennard-Jones molecules. *Computer Experiments on Classical Fluids*, 159(1):98–103.

[VonNeumann and Richtmyer, 1950] VonNeumann, J. and Richtmyer, R. D. (1950). A method for the numerical calculation of hydrodynamic shocks. *Journal of Applied Physics*, 21:232–237.

[W3C, 2014] W3C (2014). State chart xml (scxml): State machine notation for control abstraction.

[Whitcomb, 1991] Whitcomb, J. D. (1991). Iterative global/local finite element analysis. *Computers and Structures*, 40(4):1027–1031.

[Wriggers, 2002] Wriggers, P. (2002). *Computational Contact Mechanics*. John Wiley and sons, ltd.

[Wriggers and Simo, 1985] Wriggers, P. and Simo, J. C. (1985). A note on tangent stiffness for fully nonlinear contact problems. *Numerical Methods in Biomedical Engineering*, 1(5):199–203.

[Wriggers et al., 1990] Wriggers, P., VuVan, T., and Stein, E. (1990). Finite element formulation of large deformation impact-contact problems with friction. *Computers and Structures*, 37(3):319–331.

[Xiao and Belytschko, 2004] Xiao, S. P. and Belytschko, T. (2004). A bridging domain method for coupling continua with molecular dynamics. *Computer Methods in Applied Mechanics and Engineering*, 193:1645–1669.

[Xu and Belytschko, 2008] Xu, M. and Belytschko, T. (2008). Conservation properties of the bridging domain method for coupled molecular/continuum dynamics. *International Journal for Numerical Methods in Engineering*, 76:278–294.

[Zienkiewicz and Taylor, 2000] Zienkiewicz, O. C. and Taylor, R. L. (2000). *The Finite Element Method*. Butterworth Heinemann.

[Zienkiewicz and Xie, 1991] Zienkiewicz, O. C. and Xie, Y. M. (1991). A simple error estimator and adaptive time stepping procedure for dynamic analysis. *Earthquake Engineering and Structural Dynamics*, 20(9):871–887.

[Zohdi and Wriggers, 1999] Zohdi, T. and Wriggers, P. (1999). A domain decomposition method for bodies with heterogeneous microstructure based on material regularization. *International Journal of Solids and Structures*, 36:2507–2525.

[Zohdi et al., 2001] Zohdi, T., Wriggers, P., and Huet, C. (2001). A method of sub structuring large-scale computational micromechanical problems. *Computer Methods in Applied Mechanics and Engineering*, 190:5639–5656.