Efficient acceleration techniques for nonlinear analysis of structures with frictional contact

Anthony Giacoma

N^{O} d'ordre 2014–ISAL–0095 Année 2014

THÈSE

Efficient acceleration techniques for nonlinear analysis of structures with frictional contact

présentée devant
l'Institut National des Sciences Appliquées de Lyon

pour obtenir
le GRADE DE DOCTEUR

École doctorale :
Mécanique, Énergétique, Génie Civil, Acoustique

Spécialité :
MÉCANIQUE - GÉNIE MÉCANIQUE - GÉNIE CIVIL

par

Anthony GIACOMA
Ingénieur INSA de Lyon

Thèse soutenue le 2 octobre 2014 devant la commission d'examen

Jury

P. Alart	Professeur (LMGC, Univ. Montpellier 2)	Président du jury
P.-A. Boucard	Professeur (LMT, ENS de Cachan)	Rapporteur
F. Chinesta	Professeur (GeM, École Centrale de Nantes)	Rapporteur
P. Le Tallec	Professeur (LMS, École Polytechnique, Palaiseau)	Examinateur
P. Hild	Professeur (IMT, Univ. Paul Sabatier, Toulouse 3)	Examinateur
M. Rochette	Docteur (ANSYS France)	Examinateur
D. Dureisseix	Professeur (LaMCoS, INSA de Lyon)	Directeur de thèse
A. Gravouil	Professeur (LaMCoS, INSA de Lyon)	Directeur de thèse

LaMCoS - UMR CNRS 5259 - INSA de Lyon
20, avenue Albert Einstein, 69621 Villeurbanne Cedex, France

INSA Direction de la Recherche - Ecoles Doctorales – Quinquennal 2011-2015

SIGLE	ECOLE DOCTORALE	NOM ET COORDONNEES DU RESPONSABLE
CHIMIE	**CHIMIE DE LYON** **http://www.edchimie-lyon.fr** Sec : Renée EL MELHEM Bat Blaise Pascal 3e etage 04 72 43 80 46 Insa : R. GOURDON	M. Jean Marc LANCELIN Université de Lyon – Collège Doctoral Bât ESCPE 43 bd du 11 novembre 1918 69622 VILLEURBANNE Cedex Tél : 04.72.43 13 95 directeur@edchimie-lyon.fr
E.E.A.	**ELECTRONIQUE, ELECTROTECHNIQUE, AUTOMATIQUE** **http://edeea.ec-lyon.fr** Sec : M.C. HAVGOUDOUKIAN eea@ec-lyon.fr	M. Gérard SCORLETTI Ecole Centrale de Lyon 36 avenue Guy de Collongue 69134 ECULLY Tél : 04.72.18 60.97 Fax : 04 78 43 37 17 Gerard.scorletti@ec-lyon.fr
E2M2	**EVOLUTION, ECOSYSTEME, MICROBIOLOGIE, MODELISATION** **http://e2m2.universite-lyon.fr** Sec : Safia AIT CHALAL Bat Darwin - UCB Lyon 1 04.72.43.28.91 Insa : H. CHARLES	Mme Gudrun BORNETTE CNRS UMR 5023 LEHNA Université Claude Bernard Lyon 1 Bât Forel 43 bd du 11 novembre 1918 69622 VILLEURBANNE Cédex Tél : 06.07.53.89.13 e2m2@ univ-lyon1.fr
EDISS	**INTERDISCIPLINAIRE SCIENCES-SANTE** **http://www.ediss-lyon.fr** Sec : Safia AIT CHALAL Hôpital Louis Pradel - Bron 04 72 68 49 09 Insa : M. LAGARDE Safia.ait-chalal@univ-lyon1.fr	Mme Emmanuelle CANET-SOULAS INSERM U1060, CarMeN lab, Univ. Lyon 1 Bâtiment IMBL 11 avenue Jean Capelle INSA de Lyon 696621 Villeurbanne Tél : 04.72.68.49.09 Fax :04 72 68 49 16 Emmanuelle.canet@univ-lyon1.fr
INFOMATHS	**INFORMATIQUE ET MATHEMATIQUES** **http://infomaths.univ-lyon1.fr** Sec :Renée EL MELHEM Bat Blaise Pascal 3e etage infomaths@univ-lyon1.fr	Mme Sylvie CALABRETTO LIRIS – INSA de Lyon Bat Blaise Pascal 7 avenue Jean Capelle 69622 VILLEURBANNE Cedex Tél : 04.72. 43. 80. 46 Fax 04 72 43 16 87 Sylvie.calabretto@insa-lyon.fr
Matériaux	**MATERIAUX DE LYON** **http://ed34.universite-lyon.fr** Sec : M. LABOUNE PM : 71.70 –Fax : 87.12 Bat. Saint Exupéry Ed.materiaux@insa-lyon.fr	M. Jean-Yves BUFFIERE INSA de Lyon MATEIS Bâtiment Saint Exupéry 7 avenue Jean Capelle 69621 VILLEURBANNE Cedex Tél : 04.72.43 83 18 Fax 04 72 43 85 28 Jean-yves.buffiere@insa-lyon.fr
MEGA	**MECANIQUE, ENERGETIQUE, GENIE CIVIL, ACOUSTIQUE** **http://mega.universite-lyon.fr** Sec : M. LABOUNE PM : 71.70 –Fax : 87.12 Bat. Saint Exupéry mega@insa-lyon.fr	M. Philippe BOISSE INSA de Lyon Laboratoire LAMCOS Bâtiment Jacquard 25 bis avenue Jean Capelle 69621 VILLEURBANNE Cedex Tél :04.72 .43.71.70 Fax : 04 72 43 72 37 Philippe.boisse@insa-lyon.fr
ScSo	**ScSo*** **http://recherche.univ-lyon2.fr/scso/** Sec : Viviane POLSINELLI Brigitte DUBOIS Insa : J.Y. TOUSSAINT	Mme Isabelle VON BUELTZINGLOEWEN Université Lyon 2 86 rue Pasteur 69365 LYON Cedex 07 Tél : 04.78.77.23.86 Fax : 04.37.28.04.48 viviane.polsinelli@univ-lyon2.fr

*ScSo : Histoire, Géographie, Aménagement, Urbanisme, Archéologie, Science politique, Sociologie, Anthropologie

Remerciements

Le travail présenté dans ce mémoire est le concours de nombreux facteurs favorables. Ceux-ci ont été prodigués par plusieurs personnes auxquelles j'exprime toute ma gratitude. Mes premières pensées vont bien sûr à mes directeurs de thèse : David Dureisseix, Anthony Gravouil et Michel Rochette qui m'ont accordé leur confiance sur ce sujet, leur attention particulière et leurs encouragements indéfectibles. Durant trois ans, j'ai profité d'un encadrement exceptionnel tant sur le plan pédagogique et scientifique mais aussi humain. Je ne les remercierai jamais assez pour tout ce qu'ils m'ont apporté durant cette aventure que j'ai le devoir, à présent, de cultiver.

J'exprime aussi toute ma gratitude à Pierre Alart pour m'avoir fait l'honneur d'être président du jury, Pierre-Alain Boucard et Paco Chinesta pour avoir accepté de relire minutieusement ce tapuscrit et m'avoir apporté des remarques et commentaires bénéfiques à travers leurs rapports détaillés. Je remercie vivement aussi Patrick Le Tallec et Patrick Hild d'avoir porté leur attention à mon travail.

Vient maintenant le tour de toutes les personnes qui m'ont entouré (et certainement supporté) durant ces trois années. J'aimerais spécialement remercier mes voisins de bureau : Emmanuel Pillet (au discours engagé et passionné), Ramdane Lagha (au verbe enflammé dont toi seul a le secret), "Giiilles" Venet (mon camarade bâbord ... ça commence à faire quelques kilomètres que l'on rame) et Hassan Al-Akhras (qui m'initia aux bases canoniques de pantalons). J'ai aussi une pensée particulière pour Emmanuel Delor, Jean-Daniel Beley, Stéphane Garreau qui m'ont apporté une appréciation pragmatique de ce travail. J'ai ainsi pu prendre conscience que le mariage de la science et de sa mise en pratique au sein d'un logiciel commercial est véritablement un art à part entière. Merci aussi à Florent Galland qui m'a donné l'envie de démarrer cette aventure.

J'aimerais exprimer ma profonde affection pour l'équipe ANSYS Lyon qui m'a donné un environnement de travail enchanteur. Aussi, l'expression de mon amitié pour Ramdane Lagha, Gilles Venet, Emmanuel Delor, Laurent Gerboud, Gabriel Messager, Gianluca Mazzettino, Marion Ballage, Mathilde Gallouet ... Que les non-cités me pardonnent et soient rassurés, je pense tout autant à vous. J'ai aussi une pensée pour Clément Roux, Jéremy Marty, Alexandre Chemin : merci de votre accueil à chacun de mes passages au LaMCoS.

Pour finir, je pense à ma famille : ma tante Jeanine, mon père Robert, ma mère Yvonne, mon frère Roberto qui m'ont toujours soutenu et encouragé. Sans vous, rien n'aurait été possible.

“ Que de choses il faut ignorer pour agir ! ”

— Paul Valéry (Choses tues)

Résumé

La mécanique computationnelle est un outil incontournable pour le monde de l'ingénierie mécanique. Motivé par un désir de réalisme et soumis à un perpétuel gigantisme, les modèles numériques doivent aujourd'hui inclure des phénomènes physiques de plus en plus complexes. Par conséquence, d'importantes capacités calculatoires sont requises afin de traiter des problèmes à la fois non-linéaires mais aussi de grande taille. Pour atteindre cet objectif, il convient de développer les stations de calculs mais aussi les méthodes algorithmiques utilisées afin de résoudre efficacement ces types de problèmes. Récemment, les méthodes de réduction de modèle se révèlent comme d'excellentes options au développement d'algorithmes de résolution performants.

Le problème du contact frottant entre solides élastiques est particulièrement bien connu pour sa complexité et dont les temps de calcul peuvent devenir prohibitifs. En effet, les lois qui le régissent sont très hautement non-linéaires (non différentiables). Dans ce mémoire, nous nous proposons d'appliquer différentes méthodes de réduction de modèle (a posteriori et a priori) à ce type de problème afin de développer des méthodes de calculs accélérées dans le cadre de la méthode des éléments finis.

Tout d'abord, en se plaçant dans le cadre des petites perturbations en évolution quasi-statique, la réductibilité de diverses solutions impliquant du contact frottant est mise en évidence via leur décomposition en valeur singulière. De plus, leur contenu à échelle séparée est exhibé. La méthode non-incrémentale et non-linéaire à large incrément de temps (LATIN) est par la suite présentée. Dans un second temps et à partir des observations faites précédemment, une méthode LATIN accélérée est proposée en s'inspirant des méthodes multigrilles non-linéaires de type "full approximation scheme" (FAS). Cette méthode s'apparente en partie aux méthodes de réduction de modèle de type a posteriori. De plus, une stratégie de calcul de modes à partir d'un modèle de substitution est proposée. Par la suite, la décomposition propre généralisée (PGD) est utilisée afin de développer une méthode de résolution non-linéaire efficace reposant fondamentalement sur une approche de réduction de modèle de type a priori. Enfin, quelques extensions sont proposées telle que la résolution de problème faisant intervenir des études paramétriques, ou encore la prise en charge de non-linéarités supplémentaires telle que la plasticité.

Mot clés : contact frottant, LATIN, réduction de modèle, PGD, modèles de substitution.

Summary

Computational mechanics is an essential tool for mechanical engineering purposes. Nowadays, numerical models have to take into account complex physical phenomenons to be even more realistic and become larger and larger. As a consequence, more and more computing capacities are required in order to tackle not only non-linear problems but also large scale problems. For that purpose, both computers and numerical methods have to be developed in order to solve them efficiently. In the last decades, model reduction methods show great abilities to assign such challenges.

The frictional contact problem between elastic solids is particularly well-known for its difficulty. Because its governing laws are highly non-linear (non-smooth), prohibitive computational time can occur. In this dissertation, model reduction methods (both a posteriori and a priori approaches) are deployed in order to implement efficient numerical methods to solve frictional contact problem in the finite element framework.

First, small perturbations hypothesis with a quasi-static evolution are assumed. Then, reducibility of some frictional solutions is emphasized and discussed using the singular value decomposition. In addition, a scale separability phenomenon is enlightened. Then, the non-linear large time increment method (LATIN) is introduced. Secondly, an accelerated LATIN method is suggested by drawing an analogy between previous scale separability observations and the non-linear multigrid full approximation scheme (FAS). This accelerated non-linear solver relies essentially on the a posteriori model reduction approach. A precomputation strategy for modes relying on surrogate models is also suggested. Next, the proper generalized decomposition (PGD) is used to implement a non-linear solver relying fundamentally on an a priori model reduction method. Finally, some extensions are given to assign parametric studies and to take into account an additional non-linearity such as elastoplastic constitutive laws.

Key words : frictional contact, LATIN, model reduction, PGD, surrogate model.

Contents

Introduction

Contact mechanics is an interaction phenomenon occurring between one (auto-contact) or several bodies at their interface and is relevant to a wide range of engineering applications: from small mechanical assembly involving few solids to granular media. Such a fundamental phenomenon is governed by complex laws (non-linear and non smooth laws) and arouses strong numerical issues as far as the computational time is concerned. The scope of this dissertation focuses on structural design and mechanical assemblies involving few contacting bodies with friction (on the contrary to granular media). The finite element method will be used and the quasi-static small perturbation regime (small sliding motion) will be assumed. All in all, the herein presented works aims at solving a time dependent problem involving a non-smooth behavior localized at the boundary.

Even if the previously described framework is considered, classic non-linear solvers can lead to prohibitive computational time to solve the frictional contact problem. For that purpose, several classes of solvers were proposed such as augmented Lagrangian methods [BMP12], non-linear Gauss-Seidel methods [Jea99], Newton's methods [Ren12], as well as gradient methods [RA05] but their efficiency depends strongly on the the studied problem. Moreover, their convergence proof are difficult to provide due to the involved non-smooth laws. To address computational time issues, acceleration strategies based on multilevel computing techniques and multigrid methods were also suggested in [Gre95, LRR07]. During the last decades, model reduction techniques surge in enthusiasm to accelerate the resolution of both linear and non-linear solvers. Such methods consist in solving a problem into a reduced subspace which is expected to capture the "most dominant trends" of the solution. Usually, two classes of model reduction techniques are distinguished:

- a posteriori approach consisting in reusing prior knowledge about the solution (prior computations, surrogate models, etc.) to design the considered reduced subspace wherein the considered problem is solved;
- a priori approach consisting in designing on the fly a suited reduced subspace.

Both of these methods result in cheap reduced basis computations sparing hence computational work. In this dissertation, these two model reduction approaches will be tackled to implement efficient non-linear solvers for frictional contact problem.

"Dominant trends" of the solution can be identified by computing the proper orthogonal decomposition (POD) or its singular value decomposition (SVD). Originally, such

decomposition were designed to capture "coherent structures" in a turbulent flow [HLB98]. Nowadays, they are precious tools to design reduced basis that capture the most relevant and the most contributory components (or modes) of the solution. Thereby, one can expect to write the solution with a sample of few modes (reducibility property). In our framework, it will be pointed out that the concept of dominant trends is still meaningful throughout the different scales of the problem (structural scale and contact scale).

At a first glance, a priori model reduction techniques seems to be inappropriate to solve frictional contact problems. Indeed, if one uses an incremental approach to solve the time-dependent non-linear problem (for instance the Newton-Raphson procedure), an a priori approach is expected to catch the dominant modes of the solution from first time steps and guess a relevant reduced subspace. Nonetheless, the non-smooth behavior of involved laws does not allow such an achievement. To circumvent this issue, the non-increment large time increment (LATIN) [Lad99] method will be introduced and will play a key role in implementing accelerated strategies. It consists in a global space-time approach.

This typescript is outlined as follows:

- In a first chapter, an overview of reduction methods is given. Then, a brief theoretical background of POD and SVD is recalled. Afterwards, the reference frictional contact problem is posed by starting from the continuous model up to its discretization with the finite element method. Next, the SVD is applied on some frictional solutions to exemplify the concept of space-time reducibility and to make sense of "dominant trends" (*i.e.* identify scales of the problem). Finally, the LATIN method is introduced and formulated to solve the frictional contact problem.

- In a second chapter, an accelerated version of the LATIN method is proposed. It is based on an analogy with multigrid methods. A computational strategy is also suggested in order to guess in an inexpensive way dominant trends of the sought solution using surrogate models.

- In a third chapter, an a priori approach is introduced to solve the frictional contact problem. Such a method is able to solve the frictional problem without a priori knowledge about the solution in a computed on-the-fly subspace. It will be pointed out that such a strategy shows a quasi-optimal property in the sense that the designed subspace is close to the most suited one.

- In a fourth chapter, some development paths are given as extensions for the proposed strategies. First, parametric studies are tackled using reduced order model (ROM) techniques and, then, proper generalized decomposition (PGD). Second, a material non-linearity (plasticity) is tackled. An introductory LATIN formulation embedding an a priori reduced basis strategy is proposed to solve this new problem.

Along this dissertation, numerical examples will be exhibited to illustrate and to discuss performances of the suggested strategies.

Chapter 1

Reduction methods applied to frictional contact problem

Abstract

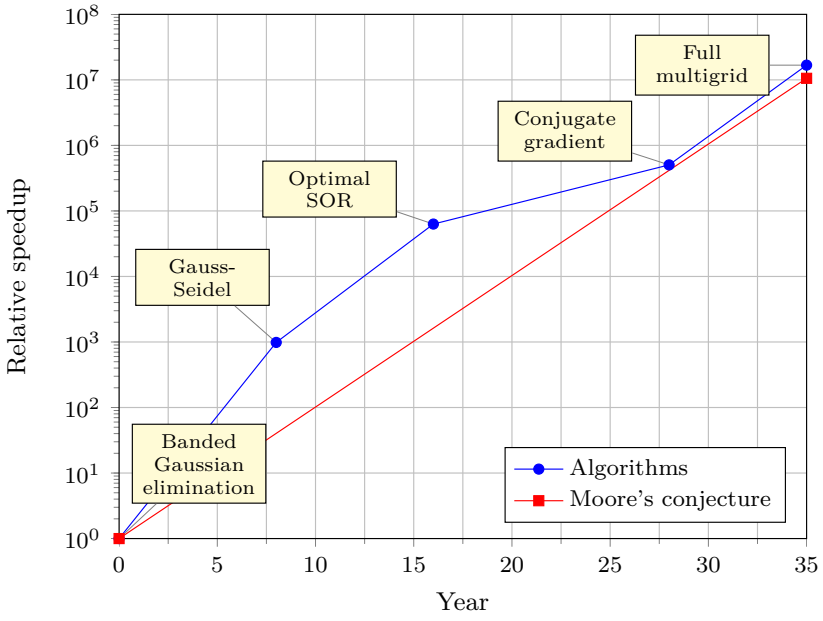
Computational contact mechanics is a challenging issue. Even if computer architectures or numerical methods greatly improve, simulations of those numerical problems lead to strong numerical difficulties and especially long computational times. Nowadays, reduction methods surge in enthusiasm. This class of numerical methods shows great abilities to assign this issue. This first chapter sets the framework of this dissertation. For that purpose, main definitions and techniques related to reduction methods are given. Afterwards, the frictional contact problem is posed. Next, the reducility of a frictional contact solution and its multiscale content is exemplified and discussed in a global space-time framework. Even if application of reductions methods to such problems seems to be inappropriate, those observations seem to open us promising horizons. Those techniques will be essentially deployed to develop efficient non-linear solvers for frictional contact problems.

Contents

1.1 The computational speedup race

Computer sciences provide attractive and efficient tools for various mathematical modeling fields and industrial purposes. Since 1960, computer simulations have become an essential part of physics, engineering, industrial development and in even more unexpected field such as financial prediction, social science, biology... Last decades, evolution of computer and supercomputer architectures provides a first response to more and more demanding communities and break the petaflop barrier in 2008 with the IBM Roadrunner. This hardware evolution with the HPC[1] was intuited by the so-called Moore's law [Moo65].

This overall computational improvement is also due to a simultaneous evolution of numerical methods [GKC+09]. These last acted on an equal level as far as the speedup was gained. To illustrate it, figure 1.1 shows the relative speedup achieved by the evolution of algorithms to solve the linear Poisson equation.



Method	Storage	Flops
Gaussian elimination (banded)	n^5	n^7
Gauss-Seidel	n^3	$n^5 \log(n)$
Optimal SOR	n^3	$n^4 \log(n)$
Full multigrid	n^3	n^3

Figure 1.1 – Data collected from [CDJGK03]. Memory requirements and performance to solve the linear 3D Poisson equation on a uniform n^3 ($n = 64$) grid with the finite difference method. The Moore's conjecture stating that performances is doubled every 18 months is also plotted.

We understand that the global computational acceleration is a synergy between not

[1] High Performance Computing

only hardware capabilities but also numerical methods.

Last decades, reduction methods spur a strong interest to design efficient algorithms. Such techniques shows great abilities to accelerate solutions for numerous linear and non-linear problems and provide well suited strategies for parametric studies. On the other hand, frictional contact problems are well known for their toughness and raise convergence issues. However, application of reduction methods seems to be inappropriate due to the non-smooth properties of such problems. This dissertation tackles this challenging issue and aims to provide efficient solvers based on reduction methods to accelerate the computational time of frictional contact problems.

1.2 Introduction to reduction methods

1.2.1 Numerical modeling

Before obtaining a workable numerical model, several modeling steps occur. These generate three kinds of error which has to be estimated to qualify the relevancy of the final result. In this dissertation, only the numerical aspect is tackled. Hence, it is assumed that the only numerical error accounts. A brief overview of the different modeling phases is hereinafter given.

Continuous modeling The physical model is put into equations. Thus, one has to choose the most suited mathematical models and governing laws to fit it at best. This first choice induces a modeling error.

Discretized model Generally speaking, the mathematical model is formulated with partial differential equations (PDEs) and can not be formally solved. Then, numerical methods are used (*e.g.* finite element method, finite difference method, etc) and characteristic variables are discretized. The time is also discretized by using time schemes according to quasi-static or dynamic formulations (*e.g.* θ-method [SP91, BGH00], Newmark based methods [CH93], etc). Then, a discretization error results. An upper bound for this error is generally provided and depends on a characteristic length of the discretization [Ern04, LP05]. Hereinafter, we consider this numerical model as the reference.

Numerical resolution Once the model is discretized, the problem is generally cast into an algebraic formulation. The resulting linear or non-linear system may be large and suited solvers provide the desired solution. But, it is affected by a numerical error involved solver parameters, computer arithmetic, round-off errors, etc. This stage is generally time consuming depending on the size of the numerical problem and may arouse memory storage issues. In this dissertation, reduction methods are suggested to accelerate and enhance the resolution process.

1.2.2 Reduction methods

The finite element method (FEM) is an efficient and widespread numerical method for spatial discretization of structural mechanics problems. Generally speaking, it provides an algebraic formulation of the problem which has to be solved into a space $\mathcal{E}$ of dimension n (*i.e.* the number of degrees of freedom) for some m discrete time steps. The space $\mathcal{E}$ is spanned by the canonical basis composed of n unitary vectors (equaling 1 at a given degree of freedom and 0 elsewhere). Reduction methods consist in finding an appropriate basis composed of $r < n$ vectors (the so-called reduced basis) spanning a subspace of $\mathcal{E}$ wherein the problem will be solved. This way, computational work may be spared. For instance, in modal analysis of mechanical systems, the modal reduction method consists in choosing the subspace spanned by the first normal modes of the studied body (modal truncation method). Similarly, the Craig and Bampton method [BC68] consists in choosing other specific modes according to given boundary conditions and external loads. A large literature shows how large is the field of applications of such techniques. Nevertheless, finding a basis providing both an attractive dimensional reduction and a relevant solution is challenging. As the solution is prescribed to a given subspace an additional error may also occur. It depends of course on the relevancy of the considered subspace and could be estimated [LC11]. Generally speaking, reduction methods are distinguished between a posteriori approaches and a priori approaches.

First, an a posteriori approach consists in prescribing a basis spanning a subspace before performing computations. Then, equations of the numerical model are projected using *e.g.* a Galerkin method. Finally, sought solution is computed (online phase) using the resulting reduced order model (ROM). The computed solution belongs to the prescribed subspace and its accuracy depends strongly on the relevancy of the considered reduced basis [HBN13, HBN12]. It generally requires a low computational effort and a such strategy is particularly suited for on-board computations and parametric problems. Both linear and non-linear problems can be tackled [KGAB11, KV01, RCCA06, AZF12b]. Nevertheless, to design a working subspace some prior knowledge about the solution (called snapshots) are required. These prerequisites can be obtained from prior computations, analytic solutions, approximation of the solution on surrogate models, etc. Then, a relevant reduced basis for them is computed using for instance a proper orthogonal decomposition or a singular value decomposition. This preliminary phase is called offline phase and may be expensive.

On the contrary, a priori approach does not require prior knowledge about the solution. The reduced basis is computed and adapted on-the-fly during the resolution process. This dissertation will focus on the widespread a priori reduced basis method called proper generalized decomposition (PGD) [LPN10, CAC10, CLC11, BGA13, CKL14]. It is a general method tackling various problems and consists in searching the solution into a

low rank approximation (or separated form):

$$u(x,t) \simeq \sum_{k=1}^{p} f_k(x) g_k(t) \tag{1.1}$$

A such representation for the solution provides several advantages as far as the computational work is engaged and the storage memory is required. Nonetheless, drawbacks of this approach rely on the sustainability of the low rank format for the solution (lack of orthogonal properties on the basis). It is generally expected that a short expansion (*i.e.* p is small) suffices to obtain a good approximation for the solution. For some applications such as transient dynamics [BGA13] or problems involving moving boundaries, the low rank approximation is seemingly inappropriate and advanced other features are required.

Generally, efficient reduction methods involves a mixed approach between a posteriori and a priori. For instance, given a reduced basis, some calculations are performed on a ROM but results obtained do not satisfy a certain level of accuracy (a posteriori approach). Then, the considered reduced basis can be enriched to span an improved solution. A priori methods are able to fulfill this goal. To take another example, instead of initializing a priori methods from scratch, prior knowledge about the solution could be reused providing a first guess for the subspace to compute on-the-fly. That's why generally speaking, efficient strategies involve both a posteriori and a priori methods in complementary roles [Ryc05, KGAB11, Gal11, GGMR11, HBN13].

To clarify this section, a summarizing workflow is proposed on figure 1.2. In this dissertation, a posteriori (LATIN-FAS and LATIN-ROM) and a priori (LATIN-PGD) strategies are tackled. But as mentioned before, except for the suggested LATIN-ROM (pure a posteriori approach), both approaches are mixed. In this dissertation, reduction methods are deployed principally to design efficient solvers for frictional contact problems. Parametric studies is here not widely tackled and only development paths are given in chapter 4.

1.3 Reduced basis design

This section tackles compression issues and provides efficient tools to extract dominant trends of an amount of data. In our framework, considered data types are evaluation functions also called snapshots obtained from computations (*e.g.* a generalized displacement field $\mathbf{u}$). For a sake of memory usage and to address storage issues, a compression method aims at computing a basis for the considered amount of data (*e.g.* in our case, it would be a basis of functions spanning the snapshots). For that purpose, a Gram-Schmidt process could be used for instance. By doing so, the different trends of snapshots can be identified and redundancy among snapshots is eliminated. Nevertheless, such and orthogonalization process does not provide the most dominant directions for snapshots. To achieve it, specific methods such as the the proper orthogonal decomposition (POD) and singular value

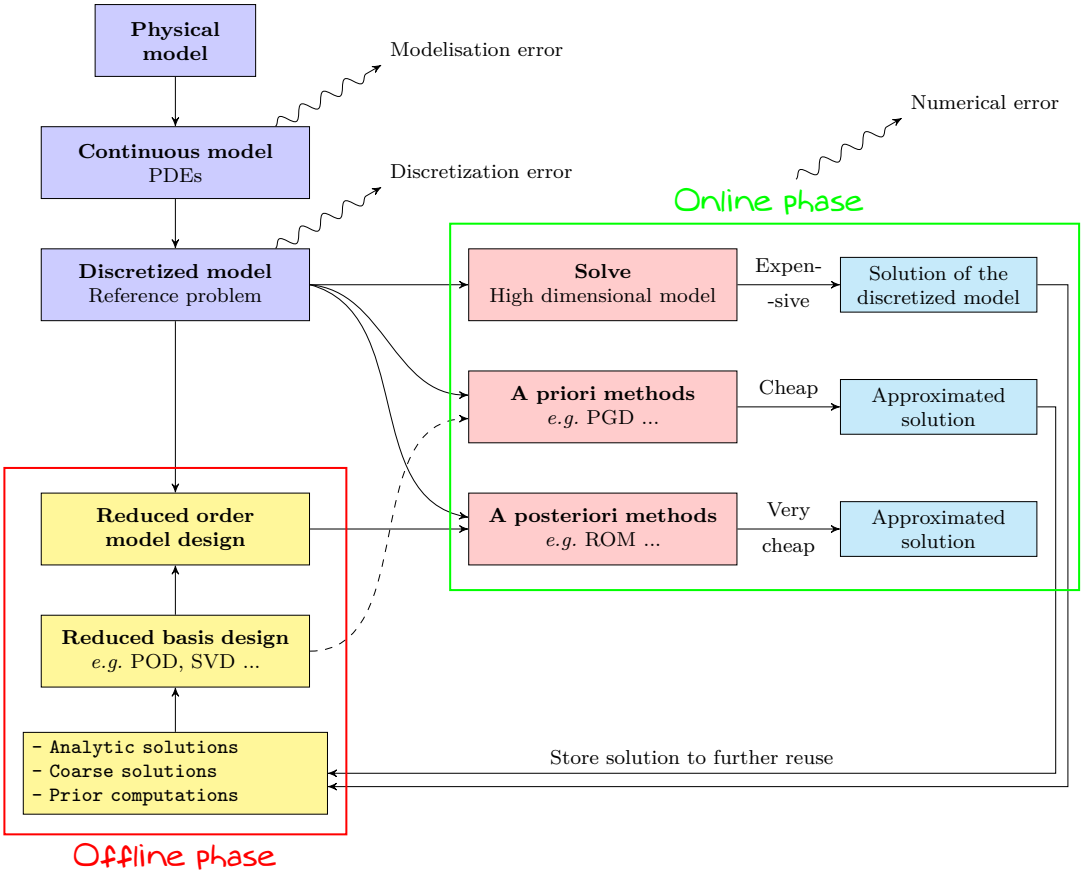


Figure 1.2 – Reduction methods and numerical resolution.

decomposition (SVD) are hereinafter presented. These methods originate mainly from the statistical field (principal component analysis, Karhunen-Loève decomposition, etc.) and aims to provide a reduced basis composed of most dominant vectors for snapshots.

1.3.1 The proper orthogonal decomposition (POD)

The POD was introduced by [HLB98] and can be related to various mathematical modeling fields. Given a set of snapshots, POD is able to compute the optimal basis according to a given norm. In other words, vectors of the POD basis are sorted according to their suitability to snapshots arising the optimality property for the POD basis. For instance, the first empirical modes will be the most representative in an average sense of the considered snapshots.

Continuous case

Hereinafter, a summarized definition and elements of proof for the POD are given. Interested readers can refer to [HLB98, Ber04] wherein detailed and rigorous definitions can be found. Let us define a vector-valued function $\mathbf{u}(\mathbf{X})$ defined on a space-time domain $\mathcal{D} = \Omega \times \mathbb{R}_+$. A function $\boldsymbol{\Phi} \in L^2(\mathcal{D})$ is sought such that the correlation between $\boldsymbol{\Phi}$ and $\mathbf{u}$ are maximal. In other words, function $\boldsymbol{\Phi}$ maximizes its projection on $\mathbf{u}$ and according to the canonical inner product on the space domain only denoted by $(\cdot \mid \cdot)$ and a time averaging operator denoted by $\langle\cdot\rangle$ (*e.g.* mean of the function):

$$\boldsymbol{\Phi} = \arg \max_{\substack{\boldsymbol{\Psi} \in L^2(\mathcal{D}) \\ (\boldsymbol{\Psi} \mid \boldsymbol{\Psi}) = 1}} \langle (\mathbf{u} \mid \boldsymbol{\Psi}) \rangle \tag{1.2}$$

so that $(\boldsymbol{\Phi} \mid \boldsymbol{\Phi}) = \|\boldsymbol{\Phi}\|^2 = 1$. It can be shown that the previous maximization problem can be cast into an eigenvalue decomposition (EVD) of the operator $\mathbf{R} : L^2(\mathcal{D}) \mapsto L^2(\mathcal{D})$ defined as follows:

$$\mathbf{R}(\boldsymbol{\Phi}) = \int_{\mathcal{D}} \left[\langle \mathbf{u}(\mathbf{X}) \otimes \mathbf{u}(\mathbf{Y}) \rangle \boldsymbol{\Phi}(\mathbf{Y}) \right] \mathrm{d}\mathbf{Y} \tag{1.3}$$

The sought function $\boldsymbol{\Phi}$ is the eigenfunction corresponding to the largest eigenvalue of the EVD: $\mathbf{R}(\boldsymbol{\Phi}) = \lambda \boldsymbol{\Phi}$. Thanks to properties of operator $\mathbf{R}$ (positive, linear and self-adjoint), some important remarks can be noticed:

- The EVD of $\mathbf{R}$ has a countable infinity solutions denoted by eigenvalues λ_k and eigenfunctions $\boldsymbol{\Phi}_k$.
- Eigenvalues λ_k are real and positive and can be indexed in decreasing order:

$$\lambda_1 \geqslant \lambda_2 \geqslant \cdots \geqslant \lambda_\infty \geqslant 0 \tag{1.4}$$

- Eigenfunctions $\mathbf{\Phi}_k$ form a complete orthonormal set (*i.e.* a basis for $\mathbf{u}$). In other words, denoting by δ_{ij} the Kronecker symbol:

$$\mathbf{u} = \sum_{k=1}^{\infty} a_k \mathbf{\Phi}_k(\mathbf{X}) = \sum_{k=1}^{\infty} (\mathbf{u} \mid \mathbf{\Phi}_k)\mathbf{\Phi}_k(\mathbf{X}) \quad \text{with} \quad (\mathbf{\Phi}_i \mid \mathbf{\Phi}_j) = \delta_{ij} \tag{1.5}$$

- Accounting the decreasing indexing for λ_k, the POD basis is the basis formed by the first K eigenfunctions $\mathbf{\Phi}_k$. These functions are called POD modes or empirical modes.

- Among all basis of K functions denoted $(\mathbf{\Psi}_k)_1^K \in L^2(\mathcal{D})$, the POD basis denoted by $(\mathbf{\Phi}_k)_1^K \in L^2(\mathcal{D})$ is the optimal one. This is the optimality property:

$$\left\langle \left\| \mathbf{u} - \sum_{k=1}^{K} (\mathbf{u} \mid \mathbf{\Phi}_k)\mathbf{\Phi}_k \right\|^2 \right\rangle \leqslant \left\langle \left\| \mathbf{u} - \sum_{k=1}^{K} (\mathbf{u} \mid \mathbf{\Psi}_k)\mathbf{\Psi}_k \right\|^2 \right\rangle \tag{1.6}$$

Discrete case

Instead of considering a vector-valued function, we introduce snapshots which are evaluations of $\mathbf{u}(\mathbf{X})$ for certain values $\mathbf{X}$ (*e.g.* numerical evaluations, experimental measures). Previous properties can be easily transposed to the present discrete case:

- The corresponding EVD has $\mathcal{K} = \dim\left[\text{span}\Big(\mathbf{u}(\mathbf{X}_i)\Big)\right]$ real and positive eigenvalues.
- Eigenvalues λ_k can be indexed in decreasing order:

$$\lambda_1 \geqslant \lambda_2 \geqslant \cdots \geqslant \lambda_{\mathcal{K}} \geqslant 0 \tag{1.7}$$

- Eigenvectors $\mathbf{\Phi}_k$ form a complete orthogonal set.
- Each snapshot $\mathbf{u}(\mathbf{X})$ can be written as a linear combination of vectors $\mathbf{\Phi}_k$.
- Each vector $\mathbf{\Phi}_k$ can be written as a linear combination of snapshots $\mathbf{u}(\mathbf{X})$.
- Accounting the decreasing indexing for λ_k, the POD basis is the basis formed by the first K vectors $\mathbf{\Phi}_k$.
- The optimality property pertains.
- As POD modes can be written as a linear combination of snapshots, if snapshots meet homogeneous Dirichlet boundary conditions, then POD modes respect also individually homogeneous boundary conditions.

In practice, to compute the POD basis of a given set of snapshots, an inner product and a time average operator as to be chose. Then, an EVD has to be solved.

1.3.2 The singular value decomposition (SVD)

Let us define a snapshot matrix $\mathbf{U}$. For instance, in the finite element framework, generalized displacements (n degrees of freedom) for all time steps $t_{0\leqslant k\leqslant m}$ can be cast into $\mathbf{U}$ as follows:

$$\mathbf{U} = \begin{bmatrix} u_1(t_0) & u_1(t_1) & \cdots & u_1(t_m) \\ u_2(t_0) & u_2(t_1) & \cdots & u_2(t_m) \\ \vdots & \vdots & \ddots & \vdots \\ u_n(t_0) & u_n(t_1) & \cdots & u_n(t_m) \end{bmatrix} = \begin{bmatrix} \mathbf{u}(t_0) & \mathbf{u}(t_1) & \cdots & \mathbf{u}(t_m) \end{bmatrix} \tag{1.8}$$

$\mathbf{U} \in \mathbb{R}^{n\times m}$ is a real rectangular matrix and $\mathbf{u}(t_k)$ are snapshots of the generalized displacement field. According to the singular value theorem [EY36], $\mathbf{U}$ can be factorized using SVD:

$$\mathbf{U} = \boldsymbol{\Upsilon}\boldsymbol{\Sigma}\boldsymbol{\Phi}^T = \begin{bmatrix} \boldsymbol{\Upsilon}_1 & \cdots & \boldsymbol{\Upsilon}_n \end{bmatrix} \begin{bmatrix} \sigma_1 & 0 & \cdots & 0 \\ 0 & \sigma_2 & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & \sigma_r \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{\Phi}_1^T \\ \vdots \\ \boldsymbol{\Phi}_m^T \end{bmatrix} \tag{1.9}$$

with $r = \min(n, m)$, $\boldsymbol{\Upsilon} \in \mathbb{R}^{n\times m}$ is an unitary matrix containing left-singular vectors, $\boldsymbol{\Phi} \in \mathbb{R}^{m\times m}$ an unitary matrix containing right-singular vectors and $\boldsymbol{\Sigma} \in \mathbb{R}^{n\times m}$ containing positive singular values σ_i in decreasing amplitudes. This decomposition is unique up to an arbitrary sign for pair $(\boldsymbol{\Upsilon}_i, \boldsymbol{\Phi}_i)$. The SVD factorization (1.9) can be rewritten into the following rank one expansion:

$$\mathbf{U} = \sum_{k=1}^{r} \sigma_i \boldsymbol{\Upsilon}_i \boldsymbol{\Phi}_i^T \tag{1.10}$$

Snapshots $\mathbf{u}(t_k)$ can be written as a linear combination of left singular vectors $\boldsymbol{\Upsilon}_i$ which coordinates $\boldsymbol{\Phi}_i(t_k)$ are amplified by singular values σ_i:

$$\mathbf{u}(t_k) = \sum_{k=1}^{r} \sigma_i \boldsymbol{\Upsilon}_i \boldsymbol{\Phi}_i(t_k) \tag{1.11}$$

Given a matrix $\mathbf{U} \in \mathbb{R}^{n\times m}$ whose its general entries is denoted by u_{ij}, the matrix p-norm is defined as follows:

$$\|\mathbf{U}\|_p = \left[\sum_{i=1}^{n} \sum_{j=1}^{m} |u_{ij}|^p \right]^{\frac{1}{p}} \tag{1.12}$$

The above norm has to be distinguished form matrix induced norms. For $p = 2$, the Frobenius norm is defined and is denoted by $\|\mathbf{U}\|_F$. Taking into account only the $K \leq r$

first singular values of the SVD expansion of $\mathbf{U}$ allows to define a low rank approximation of $\mathbf{U}$ denoted by $\check{\mathbf{U}}$ and yields to:

$$\check{\mathbf{U}} = \sigma_1 \boldsymbol{\Upsilon}_1 \boldsymbol{\Phi}_1^T + \sigma_2 \boldsymbol{\Upsilon}_2 \boldsymbol{\Phi}_2^T + \cdots + \sigma_K \boldsymbol{\Upsilon}_K \boldsymbol{\Phi}_K^T \tag{1.13}$$

According to the Eckart-Young's low rank approximation theorem [EY36], $\check{\mathbf{U}}$ is the best approximation of rank K of $\mathbf{U}$ according to the Frobenius' norm. Moreover, if $K = r$ then $\check{\mathbf{U}} = \mathbf{U}$. In other words, considering the Frobenius norm, the SVD provides the discrete POD of $\mathbf{U}$. Moreover, it can be easily shown that the relative error between the snapshot matrix $\mathbf{U}$ and its SVD approximation $\check{\mathbf{U}}$ is given by:

$$\mathcal{E}(\mathbf{U}) = \frac{\|\mathbf{U} - \check{\mathbf{U}}\|_F}{\|\mathbf{U}\|_F} = \frac{\|\mathbf{U} - \sum_i^K \boldsymbol{\Upsilon}_i \boldsymbol{\Upsilon}_i^T \mathbf{U}\|_F}{\|\mathbf{U}\|_F} = \sqrt{\frac{\sum_{i=K+1}^r \sigma_i^2}{\sum_{i=1}^r \sigma_i^2}} \tag{1.14}$$

For data compression issues, SVD is particularly interesting. Indeed, instead of storing the whole $n \times m$ entries of the snapshot matrix $\mathbf{U}$, only the first K pairs of vectors is retained representing $K(m+n)$ entries. In order to be efficient as far as the memory is concerned, we deduce the following criterion:

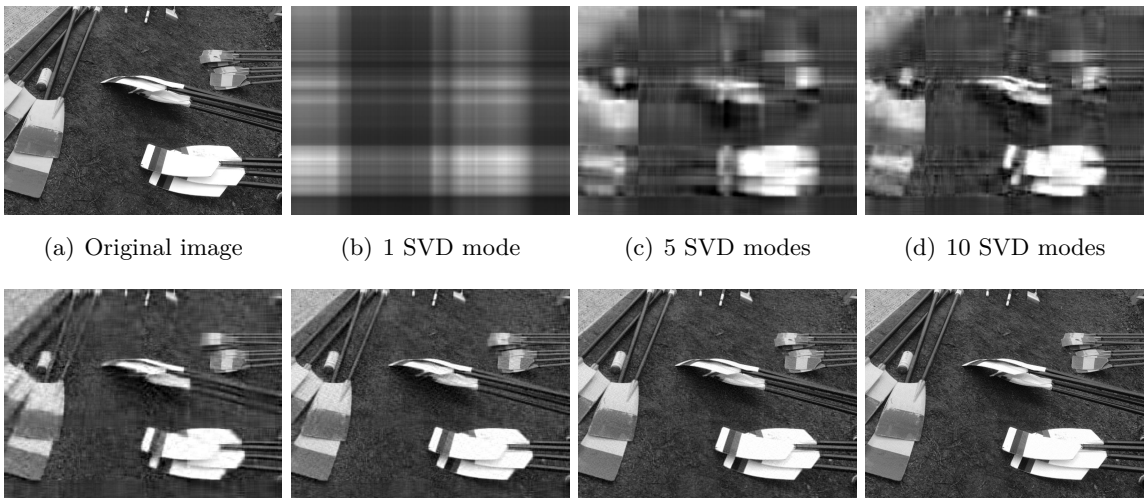
$$K(m+n) \leqslant nm \quad \Leftrightarrow \quad K \leqslant \frac{nm}{(m+n)} \tag{1.15}$$

Generally, K is chosen in such a way that a great compression is gained as far as the amount of data to store is concerned and the corresponding low rank approximation is sufficiently accurate (*e.g.* $\mathcal{E}(\mathbf{U}) < 0.1$).

1.3.3 Application to image compression

To illustrate the SVD compression ability, we propose to compress a grayscale picture sizing $n \times m = 800 \times 600$ pixels. Then a snapshot matrix, say $\mathbf{U} \in \mathbb{R}^{m \times n}$ whose entry is the grayscale value for a given pixel, can be factorized with the SVD. Consequently, an amount of 600 modes are computed. An efficient compression leads to choose p modes such that $p < \frac{mn}{m+n} = 342$. Considering only given K first SVD modes, we are able rebuilt an approximated or compressed picture (see figure 1.3).

Depending the desired rendering, 50 to 100 modes are seemingly sufficient to get a compressed but qualitative picture.



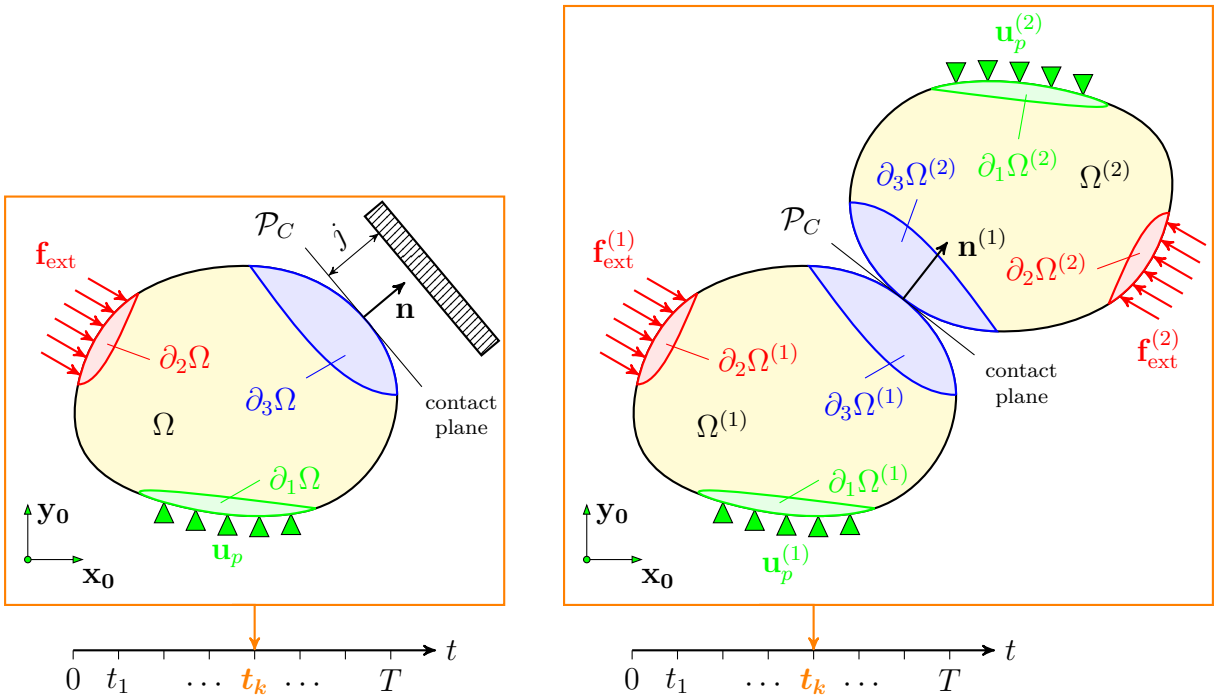
(a) Original image (b) 1 SVD mode (c) 5 SVD modes (d) 10 SVD modes

(e) 25 SVD modes (f) 50 SVD modes (g) 100 SVD modes (h) 200 SVD modes

Figure 1.3 – Original photo (rowing oars) rebuilt with SVD modes.

1.4 Frictional contact problems

1.4.1 Reference problem and governing equations



(a) Flexible vs. rigid

(b) Flexible vs. flexible

Figure 1.4 – Problem set.

A flexible body is contacting a rigid basement (figure 1.4). We define Ω, the domain occupied by this solid whose boundary is $\partial\Omega$ at time $t \in [0, T] \subset \mathbb{R}$. This boundary can be split into three complementary parts:

- $\partial_1\Omega$: the part of the boundary where displacements $\mathbf{u}_\text{p}$ are prescribed;
- $\partial_2\Omega$: the part of the boundary where external loads $\mathbf{f}_\text{ext}$ are prescribed;
- $\partial_3\Omega$: the part of the boundary where contact conditions may occur (potential contact interface).

For multibody problems, all of above quantities are affected by a superscript $^{(i)}$ according to bodies indexing. For a sake of clarity, this superscript is dropped off for single flexible body problem.

1.4.2 Strong formulation of governing equations

Linear behavior

Linear assumptions are assessed (small perturbations and strains, homogeneous-isotropic-linear-elastic material, small slidings). External loads are time-dependent and a quasi-static regime is assumed. The problem can be set as follows: find the displacement field and the Cauchy's stress field denoted respectively by $\mathbf{u}(\mathbf{x},t)$ and $\boldsymbol{\sigma}(\mathbf{x},t)$ satisfying:

- kinematic admissibility:

$$\mathbf{u}(\mathbf{x},t) \in \mathcal{U}^{[0,T]} \quad \text{and } \mathcal{U}^{[0,T]} = \left\{ \mathbf{u} \in \mathcal{H}^1(\Omega) \text{ with } \mathbf{u} \mid_{\partial_1\Omega\times[0,\ T]} = \mathbf{u}_p \right\} \tag{1.16}$$

  the strain-displacement relationship with linear assumptions reads:

$$\forall(\mathbf{x},\ t) \in \Omega \times [0,\ T] \ : \quad \boldsymbol{\varepsilon}(\mathbf{u}) = \tfrac{1}{2}(\boldsymbol{\nabla}\mathbf{u} + \boldsymbol{\nabla}^T\mathbf{u}) \tag{1.17}$$

  and the trace of the displacement field on the contact interface is $\mathbf{v} = \mathbf{u} \mid_{\partial_3\Omega\times[0,\ T]}$.

- static admissibility: $\boldsymbol{\sigma}(\mathbf{x},t)$ is balanced with external force and contact forces λ, *i.e.*

$$\begin{cases} \forall(\mathbf{x},\ t) \in \partial_2\Omega \times [0,\ T] : & \boldsymbol{\sigma}\ \mathbf{n} = \mathbf{f}_{\text{ext}} \\ \forall(\mathbf{x},\ t) \in \partial_3\Omega \times [0,\ T] : & \boldsymbol{\sigma}\ \mathbf{n} = \boldsymbol{\lambda} \\ \forall(\mathbf{x},\ t) \in \Omega \times [0,\ T] : & \mathbf{div}(\boldsymbol{\sigma}) = \mathbf{0} \end{cases} \tag{1.18}$$

  where $\mathbf{n}$ is the local outward normal vector of the considered domain.

- constitutive laws:
  - Hooke law for elasticity:

$$\forall(\mathbf{x},\ t) \in \Omega \times [0,\ T] \ : \quad \boldsymbol{\sigma} = \mathbb{K}\boldsymbol{\varepsilon}(\mathbf{u}) \tag{1.19}$$

    where $\mathbb{K}$ is the linear elasticity Hooke operator.
  - Signorini condition and Coulomb frictional law for contact. With $\mathbf{v}$ and $\boldsymbol{\lambda}$, the traces of displacement field and contact force field (both normal and tangential) on the contacting interface $\partial_3\Omega$, these conditions can be written formally as:

$$\mathcal{R}(\mathbf{v},\boldsymbol{\lambda}) = 0 \tag{1.20}$$

To be more precise for the frictional behavior, fields at the contacting interface $\partial_3\Omega$ can be split into a normal and a tangential part as:

$$\forall(\mathbf{x},\ t) \in \partial_3\Omega \times [0,\ T] : \quad \begin{cases} \mathbf{v}(\mathbf{x},t) = u_N\mathbf{n} + \mathbf{u}_T \\ \boldsymbol{\lambda}(\mathbf{x},t) = \lambda_N\mathbf{n} + \boldsymbol{\lambda}_T \end{cases} \tag{1.21}$$

$\mathcal{R}$ governs the contact behavior at the interface $\forall(\mathbf{x},\ t) \in \partial_3\Omega \times [0,\ T]$.

Non-linear behavior (flexible body vs. rigid obstacle)

According to Signorini conditions and Coulomb law on the displacement field and contact force field:

- Normal contact with Signorini conditions given in [Sig59],

$$\begin{aligned} g(\mathbf{u}) = u_N - j = \mathbf{u}\cdot\mathbf{n} - j \leqslant 0 \quad & \text{Non-penetration condition} \\ \lambda_N = (\boldsymbol{\sigma}\ \mathbf{n})\cdot\mathbf{n} \leqslant 0 \quad & \text{Compressive contact force} \\ \lambda_N g(\mathbf{u}) = 0 \quad & \text{Complementarity condition} \end{aligned} \tag{1.22}$$

  where j is the initial gap and $g(\mathbf{u})$ is the gap function.

- Tangential contact with Coulomb friction law [Cou85]:

$$\begin{aligned} &\|\boldsymbol{\lambda}_T\|_2 = \mu|\lambda_N| \quad \text{if sliding: } \|\boldsymbol{\delta}\mathbf{u}_T\|_2 \neq 0 \text{ and } \boldsymbol{\delta}\mathbf{u}_T = -\rho\boldsymbol{\lambda}_T \text{ with } \rho > 0 \\ &\|\boldsymbol{\lambda}_T\|_2 \leqslant \mu|\lambda_N| \quad \text{if sticking: } \|\boldsymbol{\delta}\mathbf{u}_T\|_2 = 0 \end{aligned} \tag{1.23}$$

  where $\mu \in [0\,, +\infty[$ is the friction coefficient and $\boldsymbol{\delta}\mathbf{u}_T$ is the increment of tangential displacement.

Non-linear behavior (flexible body vs. flexible body)

The previous relationships can be extended for two contacting flexible bodies. Writing them on the outward normal of the body 1 denoted by $\mathbf{n}^{(1)}$ yields to:

- Normal contact:

$$\begin{aligned} g(\mathbf{u}^{(2)}, \mathbf{u}^{(1)}) = j + (\mathbf{u}^{(2)} - \mathbf{u}^{(1)})\cdot\mathbf{n}^{(1)} \geqslant 0 \quad & \text{Non-penetration condition} \\ \boldsymbol{\lambda}^{(1)}\cdot\mathbf{n}^{(1)} = \lambda_N \leqslant 0 \quad & \text{Compressive contact force} \\ \boldsymbol{\lambda}^{(2)}\cdot\mathbf{n}^{(1)} = -\lambda_N \geqslant 0 \quad & \text{Reciprocity of contact force} \\ \lambda_N g(\mathbf{u}^{(2)}, \mathbf{u}^{(1)}) = 0 \quad & \text{Complementarity condition} \end{aligned} \tag{1.24}$$

  where j is the initial gap between the two bodies and g the gap function.

- Tangential contact with Coulomb friction law :

$$\begin{aligned} \boldsymbol{\lambda}_T^{(1)} &= -\boldsymbol{\lambda}_T^{(2)} \\ \|\boldsymbol{\lambda}_T\|_2 = \mu|\lambda_N| &\quad \text{if sliding: } \|\boldsymbol{\delta}\mathbf{u}_T^{(2/1)}\|_2 \neq 0 \text{ and } \boldsymbol{\delta}\mathbf{u}_T^{(2/1)} = -\rho\boldsymbol{\lambda}_T^{(1)} \text{ with } \rho > 0 \\ \|\boldsymbol{\lambda}_T\|_2 < \mu|\lambda_N| &\quad \text{if sticking: } \|\boldsymbol{\delta}\mathbf{u}_T^{(2/1)}\|_2 = 0 \end{aligned} \tag{1.25}$$

  where $\mu \in [0\,, +\infty[$ is the friction coefficient and $\boldsymbol{\delta}\mathbf{u}_T^{(2/1)}$ is the increment of tangential relative displacement (displacement of body 2 relatively to body 1).

1.4.3 Weak formulation

We define the following set of homogeneous kinematically admissible functions as:

$$\mathbf{u}(\mathbf{x},t) \in \mathcal{U}_0^{[0,T]} \quad \text{with} \quad \mathcal{U}_0^{[0,T]} = \left\{ \mathbf{u} \in \mathcal{H}^1(\Omega) \ \text{ and } \mathbf{u} \mid_{\partial_1\Omega\times[0,\ T]} = \mathbf{0} \right\} \tag{1.26}$$

Equilibrium (1.18) is equivalent to the following integral formulation with compatibility of contact force field and displacement field to contact conditions (1.22) and (1.23):

$$\forall \mathbf{u}^* \in \mathcal{U}_0^{[0,T]} : \int_\Omega \boldsymbol{\sigma} : \boldsymbol{\varepsilon}(\mathbf{u}^*) \mathrm{d}V - \int_{\partial_2\Omega} \mathbf{f}_{\text{ext}} \cdot \mathbf{u}^* \mathrm{d}S - \int_{\partial_3\Omega} \boldsymbol{\lambda} \cdot \mathbf{u}^* \mathrm{d}S = 0 \tag{1.27}$$

1.4.4 Semi-discretized weak formulation

Using Hooke law for $\boldsymbol{\sigma} = \mathbb{K}\boldsymbol{\varepsilon}$ and the finite element approximation for displacement field yields to the so-called semi-discretized equilibrium equation with respect to frictional contact conditions at each contacting node:

$$\forall \mathbf{u}^\star(t) : \int_0^T [\mathbf{u}^\star(t)]^T \mathbf{K}\mathbf{u}(t)\ \mathrm{d}t = \int_0^T \mathbf{u}^{*T}(t)[\mathbf{f}_{\text{ext}}(t) + \mathbf{f}_{\text{ctc}}(t)]\ \mathrm{d}t$$

$$\text{with} \quad \begin{cases} \mathbf{v}(t) = \mathbf{B}\mathbf{u}(t) \\ \mathbf{f}_{\text{ctc}}(t) = \mathbf{B}^T\boldsymbol{\lambda}(t) \\ \mathcal{R}(\mathbf{v},\boldsymbol{\lambda}) = 0 \end{cases} \tag{1.28}$$

$\mathbf{K}$ is the stiffness matrix, $\mathbf{f}_{\text{ext}}(t)$ and $\mathbf{f}_{\text{ctc}}(t)$ are generalized forces and $\mathbf{B}$ is a boolean matrix mapping the global vector of nodal values to the values on contact boundary nodes. All in all, we have to find displacement field $\mathbf{u}(t)$ (and $\mathbf{v}(t)$ is straightforwardly deduced) and contact force field $\boldsymbol{\lambda}(t)$ verifying (1.28) for all trial functions $\mathbf{u}^*(t)$.

1.4.5 Discretized weak formulation

The time interval is discretized into a sequence of time steps $t_{0\leqslant k\leqslant m}$ describing a regular time stepping (*i.e.* $t_{k+1} = t_k + \Delta t$). Then, the discretized space-time reference problem is defined as follows:

$$\begin{cases} \mathbf{K}\mathbf{u} = \mathbf{f}_{\text{ext}} + \mathbf{B}^T\boldsymbol{\lambda} \\ \mathbf{v} = \mathbf{B}\mathbf{u} \end{cases} \quad \text{with} \quad \begin{cases} \mathbf{v} = \mathbf{B}\mathbf{u} \\ \mathbf{f}_{\text{ctc}} = \mathbf{B}^T\boldsymbol{\lambda} \\ \mathcal{R}(\mathbf{v},\boldsymbol{\lambda}) = 0 \end{cases} \tag{1.29}$$

1.4.6 Challenges of frictional contact

Origin and frictional contact modeling

In 1933, Signorini posed the so called Signorini's problem: a linear elastic body resting on a rigid frictionless plane and gave unilateral contact conditions (1.22) ("Problema

con ambigue condizioni al contorno" in [Sig59]). These equations govern the interaction between two contacting bodies in the normal direction. The complementarity condition states that either the gap is positive (contact open) then contact force must be null or the gap is null (contact close) then contact forces appears. Case of both gap and contact force are null is called grazing touch. In 1963, Fichera [Fic64, Fic73] gave the proof of existence and uniqueness of the solution of this problem.

In 1699, Amontons [Amo32] states first principles of the Coulomb's frictional model:

- Friction force is proportional to the normal force.
- Frictional force depends only on material of contacting surfaces (characterizing the coefficient of friction).

In 1785, Coulomb [Cou85] added :

- Friction force is independent of the sliding velocity and resists relative lateral motion.

A frictional coefficient $\mu = 0$ corresponds to the frictionless case. Usually $\mu \leqslant 1$ and depends of the couple of involved materials, roughness, etc. Static friction and kinetic friction can be distinguished:

- Static friction with μ_s occurs between two solids in contact that are not moving relative to each other initially. This friction has to be overcome to move one of them .
- Kinetic friction with μ_k occurs between two solids in contact that are sliding relative to each other initially. In general, the force to apply in order to maintain the relative motion between these solids is less than the force to apply to overcome static friction.

Then, $\mu_s > \mu_k$. Variation of coefficient of friction is usually invoked to explain stick-slip phenomena. We assume that static friction is equivalent to kinetic friction (*i.e.* $\mu_s = \mu_k = \mu$). Generally speaking, frictional phenomenon is complex and hard to model. In the first half of the 20^{th} century, more sophisticated frictional laws with varying frictional coefficient (according to temperature, relative velocity, etc) was proposed (*e.g.* Stribeck's law). Mathematically speaking, elements to prove existence and uniqueness for a frictional Signorini's problem considering a Coulomb's model can be found in [Coc84, Kla90, RC01, BB05, Cap11].

Numerical challenges

Contact mechanics is a broad topic involving several geometrical, optimization and numerical aspects relying on strong mathematical foundations. A great coverage of computational contact mechanics with all its basic ingredients is available in [Yas13]: from contact detection to implementation of contact algorithms in a finite element software. We

will focus on the algorithmic aspect to solve frictional contact problem. Then, geometry, discretization and contact detection issues are not in the scope of this dissertation.

Frictional contact problems are strongly non-linear due to status switches (contact / no contact, sticking / sliding). Evolution of status is not regular, making such problems non-smooth and hence non-differentiable in the classic Frechet's sense. Due to this fact, specific mathematical modeling field related to the convex optimization theory is required. [Mor67, Mor76, Roc97, HUL01, CPS09]. Numerically speaking, this tough non-smooth property is challenging and specific solvers have to be designed to fulfill suited calculations.

Over the last decades, various solving methods have been developed. Contact problems can be formulated as an optimization problem subject to mathematical constraints with the augmented Lagrangian formulation. Such formulation can be solved using augmented Lagrangian methods [SL92, Lau92a, Wri95, HL99, PC99, FG00, AD08, BMP12]. Non-linear Gauss-Seidel solvers are also developed and applied to problems involving multi-contact bodies (rigid or flexible) [JAJ98, Jea99, DHK02]. But these last methods are not the most efficient to solve large systems (poor rate of convergence). Gradient methods are developed and adapted [BAV01, DF01, RA05, DS05] (with projection of the descent direction on feasible region for contact laws or active-set violation control strategy ...) in order to recover well-known performances of such methods for linear systems. But the main drawback of them is the difficulty to provide a convergence proof. In spite of the non-smooth property of contact problems, widely used Newton's methods (generalized Newton's methods) are studied [CA88, Ala97, LF00, Ren12]. Such methods proved their efficiency and robustness in large scale elasticity problems with contact and friction. Once again and due to non-smooth property of contact problems, convergence proof of generalized Newton's methods are hard to provide.

According to the Lagrange method, contact problems can be cast into a linear complementarity problem (LCP) and using specific solvers such as active-set methods, Lemke's algorithm, projected successive over relaxation (PSOR) [MDL88]. In [AP97], an LCP formulation is used to solve a frictional contact problem by faceting the Coulomb's cone. Nevertheless, this approximation is tough and not really efficient (the problem to solve becomes larger). Nowadays, LCP formulations are very well suited to frictionless problems.

In general case, constraint optimization problems can be also solved with the penalty method. Despite its easy implementation, such method does not converge to the exact solution of the reference problem [BF95] and suffers numerically of bad conditioning.

Finally, the bipotential method [DSF98, JF08] provides a framework wherein suited algorithms allowing some computer cost reduction can be applied.

In [Woh11], main results on space discretization scheme and non-linear solver for the solution of frictional contact problems are provided.

Generally speaking and even within a quasi-static context, all these non-linear solvers can lead to prohibitive time of computations. Efficiency of these solver families depends

on the complexity of the mechanical system (*e.g.* number of contacting points, see figure 1.5).

To address this computational issue, several acceleration methods can be used. First, an acceleration strategy based on multigrid methods were proposed in [AL95, LRR07]. It consists in computing cheaply corrections on coarser discretizations (grids) to accelerate the convergence rate of a non-linear solver (e.g. non-smooth Newton method) also called smoother on the finest description of the problem. Second, domain decomposition techniques (FETI-based) were also proposed for frictional contact problems [DKH+10, DKM+12]. Such techniques aims at partitioning bodies into non-overlapping sub-domains processed independently. Then, interesting scalability property can be gained. On the other hand and over the last decade, model reduction techniques surge in enthusiasm within the computational mechanics field. For both linear and non-linear problems, impressive accelerations can be obtained. Nevertheless, for frictional contact mechanics, the use of such reduced basis methods [HSW12, KP14] seem to be difficult due the non-smoothness of the constitutive laws. As proposed and claimed in [GDGR14], using a space-time approach allows to circumvent this difficulty and benefiting from computational gain of such reduced basis approaches. That's why, in this dissertation, the large time increment method [Lad99] is focused and introduced. It is closed to widespread augmented Lagrangian methods [Cha96, ADR06].

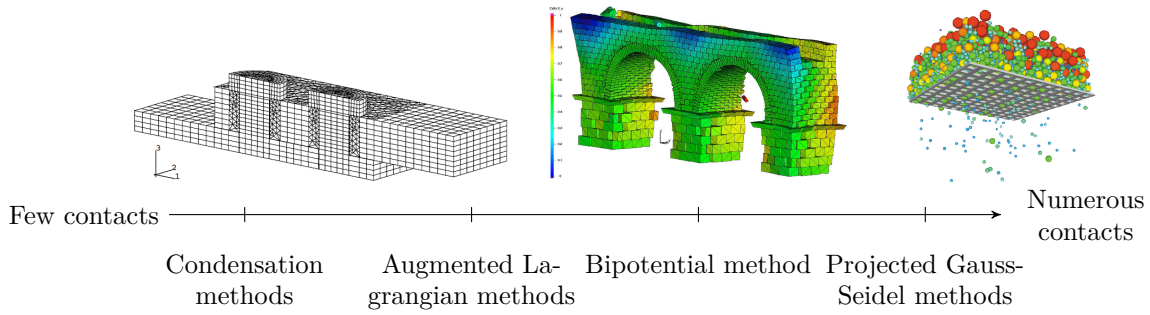


Figure 1.5 – Classification of contact problems.

1.4.7 Analogy with plasticity

Velocity formulation for frictional contact

Frictional contact conditions (1.22) and (1.23) can be equivalently rewritten using a velocity formulation. According to the viability Moreau's lemma [Mor88], unilateral contact

conditions become:

$$\begin{aligned} g(\mathbf{u}) &= \mathbf{u}\cdot\mathbf{n} - j \leqslant 0 \\ \lambda_N &= (\boldsymbol{\sigma}\ \mathbf{n})\cdot\mathbf{n} \leqslant 0 \\ \lambda_N g(\mathbf{u}) &= 0 \end{aligned} \quad \Leftrightarrow \quad \begin{cases} \text{If } g(\mathbf{u}) > 0 \text{ then} & \lambda_N = 0 \\ \text{If } g(\mathbf{u}) = 0 \text{ then} & \begin{cases} \dot{\mathbf{u}}\cdot\mathbf{n} \leqslant 0 \\ \lambda_N \leqslant 0 \\ (\dot{\mathbf{u}}\cdot\mathbf{n})\lambda_N = 0 \end{cases} \end{cases} \tag{1.30}$$

Associated flow rule

An interesting analogy can be done between standard flow rule used for metal plasticity and frictional contact laws. For isotropic elastoplastic hardening laws, a solid is said to yield once the load function f (called also plastic potential) is zeroed and defined by:

$$f(\boldsymbol{\sigma}) = J(\boldsymbol{\sigma}) - (\sigma_Y + R) \leqslant 0 \tag{1.31}$$

with $J(\boldsymbol{\sigma})$ a plastic criterion (*e.g.* Von-Mises), σ_Y the yield stress, R the hardening law. If f is negative, the solid behaves as an elastic material. The standard or associated flow rule states that once the yield limit is reached (*i.e.* $f = 0$) the plastic strain increment, and the normal to pressure dependent yield surface, have the same direction (normality condition):

$$\mathrm{d}\boldsymbol{\varepsilon}_p = \lambda\frac{\partial f}{\partial \boldsymbol{\sigma}} \ : \ \begin{cases} \text{If } f(\boldsymbol{\sigma}) < 0 \text{ then } \lambda = 0 \\ \text{If } f(\boldsymbol{\sigma}) = 0 \text{ then } \lambda \geqslant 0 \end{cases} \quad \Leftrightarrow \quad \begin{cases} f(\boldsymbol{\sigma}) \leqslant 0 \\ \lambda \geqslant 0 \\ f(\boldsymbol{\sigma})\lambda = 0 \end{cases} \tag{1.32}$$

with $\mathrm{d}\boldsymbol{\varepsilon}_p$ the plastic strain rate, $\boldsymbol{\sigma}$ the Cauchy's stress tensor and λ the plastic multiplier.

Similarly from these conditions, viability Moreau's lemma can be applied leading to the Prager's consistency condition [Pra49] stating $\mathrm{d}f = 0$ when $f = 0$:

$$\begin{cases} \text{If } f(\boldsymbol{\sigma}) < 0 \text{ then } \lambda = 0 \\ \text{If } f(\boldsymbol{\sigma}) = 0 \text{ then } \begin{cases} \mathrm{d}f \geqslant 0 \\ \lambda \geqslant 0 \\ \mathrm{d}f\lambda = 0 \end{cases} \end{cases} \tag{1.33}$$

As for frictional laws, elastoplastic governing rules are also characterized by non smooth laws (yielding or not yielding status) leading to a similar formulation. The most popular method for integrating these equations is the radial return method [OS86, Pon98]. Plasticity aspects will be detailed in the last chapter as an extension of linear frictional problems to elastoplastic frictional problems.

1.4.8 Analogy with fracture mechanics

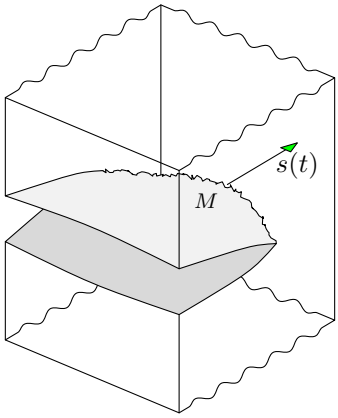


Figure 1.6 – Cracked body.

As well as plasticity, another analogy can be made with linear fracture mechanics and its maximum energy release rate (MERR, see [Gri21]) principle. Variational formulations for linear elastic fracture mechanics was proposed in [SC11] due to plasticity analogies. Let $s(t)$ be the elongation (see figure 1.6) at the crack tip at time t and its velocity denoted by $\dot{s}(t)$. First, the crack propagation is considered as irreversible, then $\forall t : \dot{s}(t) \geqslant 0$. One denotes usually by $\varphi(K_I, K_{II}, \theta)$, the maximum energy release rate criteria with K_I and K_{II} the stress intensity factors and θ the kink angle. At time t, the crack does not advance if $\varphi < 0$ or growth if $\varphi = 0$. All in all, this can be put into the following formulation:

$$\begin{cases} \dot{s}(t) \geqslant 0 \\ \varphi\Big(K_I(t), K_{II}(t), \theta(t)\Big) \leqslant 0 \\ \dot{s}(t)\varphi\Big(K_I(t), K_{II}(t), \theta(t)\Big) = 0 \end{cases} \tag{1.34}$$

The above formulation can be proved from a thermodynamical point of view [SC] and corresponds to unilateral contact condition formulation.

1.5 A posteriori empiric analysis

In this section, SVD analysis for some frictional contact problems are carried out to illustrate and exemplify arising empiric modes and enlighten a scale separability phenomenon in structural mechanics with friction. For that purpose, two numerical examples are proposed in the following. The first is a two dimensional extrusion problem involving large frictional zones relatively to the studied body. Such a problem is particularly tough. Developments given in this dissertation will be mainly illustrated on this telling example. Then, the second example is a larger three dimensional problem involving two contacting flexible bodies.

In this section, it will be assumed that solutions of suggested problem are known. These are respectively composed of the generalized displacement field and the contact force field and denoted by $\mathbf{s} = (\mathbf{u}, \boldsymbol{\lambda})$. These fields verify internal balance equation and frictional contact condition:

$$\mathbf{K}\mathbf{u} = \mathbf{f}_{\text{ext}} + \mathbf{B}^T\boldsymbol{\lambda} \quad \text{and} \quad \mathcal{R}(B\mathbf{u}, \boldsymbol{\lambda}) = 0 \tag{1.35}$$

Thanks to superposition principle, the solution is partitioned $\mathbf{s} = \mathbf{s}_0 + \mathbf{s}^* = (\mathbf{u}_0, \boldsymbol{\lambda}_0) + (\mathbf{u}^*, \boldsymbol{\lambda}^*)$ with $\mathbf{u}_0$ a kinematically admissible displacement field and $\mathbf{u}^*$ a kinematically admissible displacement field to zero. Those new fields verify:

$$\mathbf{K}\mathbf{u} = \mathbf{K}(\mathbf{u}_0 + \mathbf{u}^*) = \mathbf{f}_{\text{ext}} + \mathbf{B}^T\boldsymbol{\lambda} \quad \text{with} \quad \begin{cases} \mathbf{K}\mathbf{u}_0 = \mathbf{f}_{\text{ext}} + \boldsymbol{\lambda}_0 \\ \boldsymbol{\lambda}_0 = \mathbf{0} \\ \mathbf{K}\mathbf{u}^* = \mathbf{B}^T\boldsymbol{\lambda}^* \\ \boldsymbol{\lambda}^* = \boldsymbol{\lambda} \end{cases} \tag{1.36}$$

Solution $\mathbf{s}_0$ can be solved easily by solving a linear system and does not account for contacting boundaries. Whereas $\mathbf{s}^*$ is a "corrective solution" for $\mathbf{s}_0$ to make it compatible to frictional contact conditions. In this dissertation, suggested numerical methods are iterative algorithms for which the provided initial guess for the solution will be $\mathbf{s}_0$. Then, the corrective part $\mathbf{s}^*$ is the target field to solve. Because $\mathbf{u}^*$ is linearly bound to λ by stiffness $\mathbf{K}$, this partitioning is particularly suitable for following analysis.

The displacement field $\mathbf{u}$ (respectively $\mathbf{u}^*$) can be cast into a snapshot matrix $\mathbf{U}$ (respectively $\mathbf{U}^*$). Then, a SVD analysis will be carried out for both. From these two analyses, different quantities can be exhibited and discussed:

- From the SVD analysis of $\mathbf{U}$, a low rank approximation for the displacement field $\mathbf{u}$, say $\check{\mathbf{u}}$, can be provided. Hence, evolution of the error can be depicted according to the size of this expansion. Moreover, the elastic energy integrated over the time interval of the solution $\mathbf{u}$ defined by:

$$E = \frac{1}{2}\int_0^T \mathbf{u}(t)\mathbf{K}\mathbf{u}(t)\ \mathrm{d}t \tag{1.37}$$

  It can be distributed among empiric modes as $E = \sum_i E_i$ where

$$E_i = \frac{1}{2}{\sigma_i}^2\boldsymbol{\Upsilon}_i^T\mathbf{K}\boldsymbol{\Upsilon}_i\int_0^T \boldsymbol{\Phi}_i^T\boldsymbol{\Phi}_i\ \mathrm{d}t = \frac{1}{2}{\sigma_i}^2\boldsymbol{\Upsilon}_i^T\mathbf{K}\boldsymbol{\Upsilon}_i \tag{1.38}$$

  Accounting K modes and rebuilding $\check{\mathbf{u}}$, the fraction of captured elastic energy is $\sum_{i=1}^K E_i/E$. Nonetheless, a corresponding contact force field cannot be rebuilt. Indeed, this analysis is mainly displacement field oriented.

- From the SVD analysis of $\mathbf{U}^*$, a low rank approximation for the displacement field $\mathbf{u}^*$, say $\check{\mathbf{u}}^*$, can be provided. Moreover, from these space modes, say $\boldsymbol{\Upsilon}$, one is able

to compute associated contact force modes as follows:

$$\mathbf{K}\boldsymbol{\Upsilon} = \mathbf{F} \tag{1.39}$$

Contact force mode $\mathbf{F}$ are ensured to be localized at the contacting interface (*i.e.* $\mathbf{F} = \mathbf{B}^T\boldsymbol{\Lambda}$) due to the fact that each $\boldsymbol{\Upsilon}$ can be written as a linear combination of snapshots $\mathbf{u}^*$ verifying $\mathbf{K}\mathbf{u}^* = \mathbf{B}^T\boldsymbol{\lambda}$. This way, low rank approximations for solution fields $\mathbf{s}^*$ can be computed. Furthermore, and given $\mathbf{u}_0$, one is able to rebuilt the whole solution $\mathbf{s}$ of the problem. Errors between rebuilt fields and solution can be computed following the Frobenius norm. For instance, error for the displacement field is defined as follows:

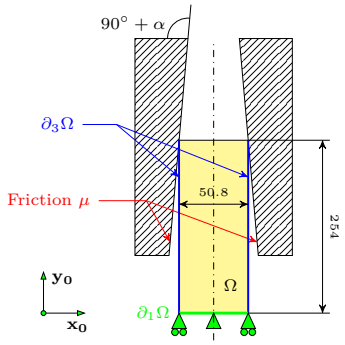
$$\mathcal{E}(\mathbf{u}) = \frac{\|(\mathbf{U}_0 + \check{\mathbf{U}}^*) - \mathbf{U}\|_F}{\|\mathbf{U}\|_F} \tag{1.40}$$

with $\mathbf{U}_0$ the snapshot matrix of $\mathbf{u}_0$ and $\check{\mathbf{U}}^*$ the snapshot matrix of $\mathbf{u}^*$. Similarly, the error $\mathcal{E}(\boldsymbol{\lambda})$ is also defined.

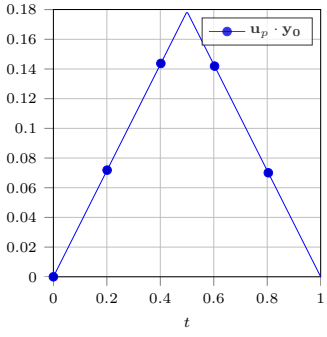
In both cases, left singular SVD modes (or empiric modes) are generalized space vectors and right singular one are time vectors. As a consequence, the SVD provides a space-time separated representation for these considered fields.

1.5.1 Extrusion problem

The first example consists in an extrusion of an elastic aluminum billet into a rigid conical die (see figure 1.7). This problem [KO88, Lau92b] is investigated assuming small perturbations even if such hypothesis is not ensured (large deformations occur). The finite element method is used and the solid is meshed with linear quadrangles elements. A displacement is prescribed in such a way that the billet is pushed into and extracted from a conical die. The solution is assumed to be known and is plotted for some time steps on figure 1.8.



(a) Numerical model

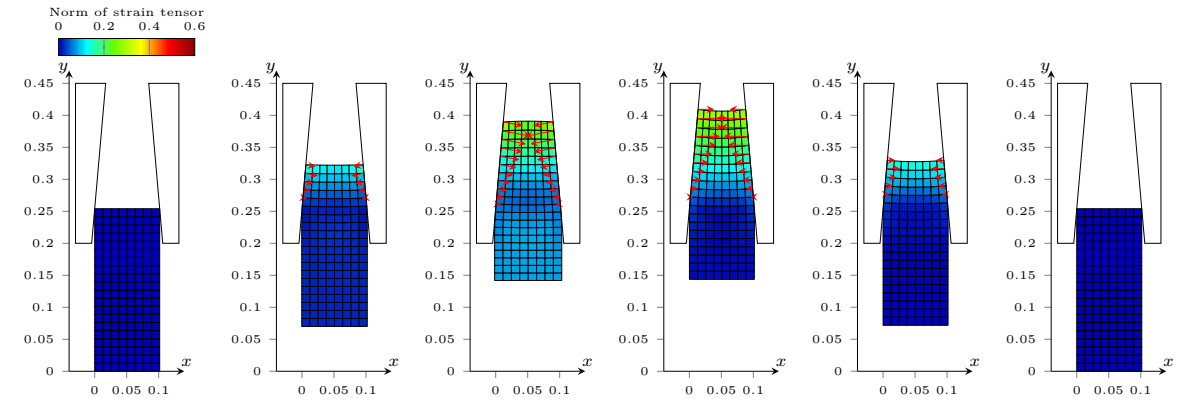


(b) Prescribed displacement

Property		Value
Young modulus	E	68.86 GPa
Poisson ratio	ν	0.32
Frictional coefficient	μ	0.1
Time interval	$[0, T]$	$[0, 1]$
Time stepping	Δt	$5 \cdot 10^{-3}$ s
Degrees of freedom	n	368
Contacting nodes	n_C	42
Angle	α	5°

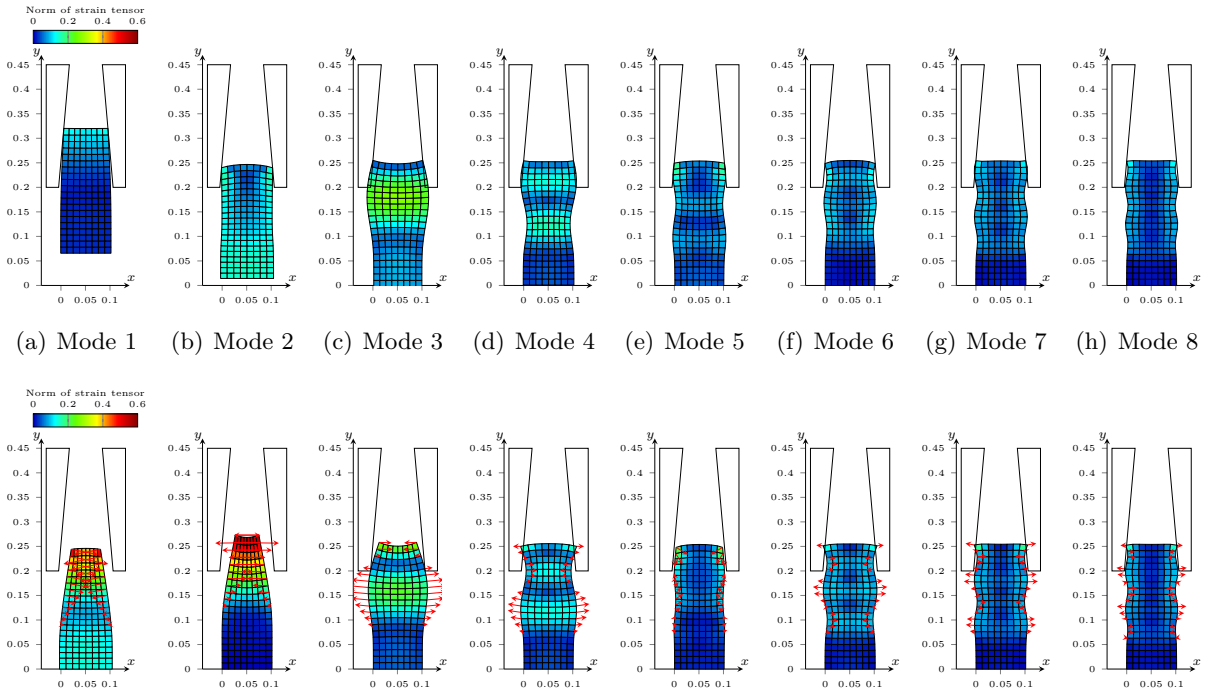
(c) Properties

Figure 1.7 – Aluminum billet pushed into conical die.



(a) t_1 (b) t_{40} (c) t_{80} (d) t_{120} (e) t_{160} (f) t_{200}

Figure 1.8 – Solution **s**: displacement and contact force fields. The color map refers to the norm of the strain tensor field $\|\boldsymbol{\varepsilon}\| = \sqrt{\boldsymbol{\varepsilon} : \boldsymbol{\varepsilon}}$ and red arrows correspond to nodal contact force.

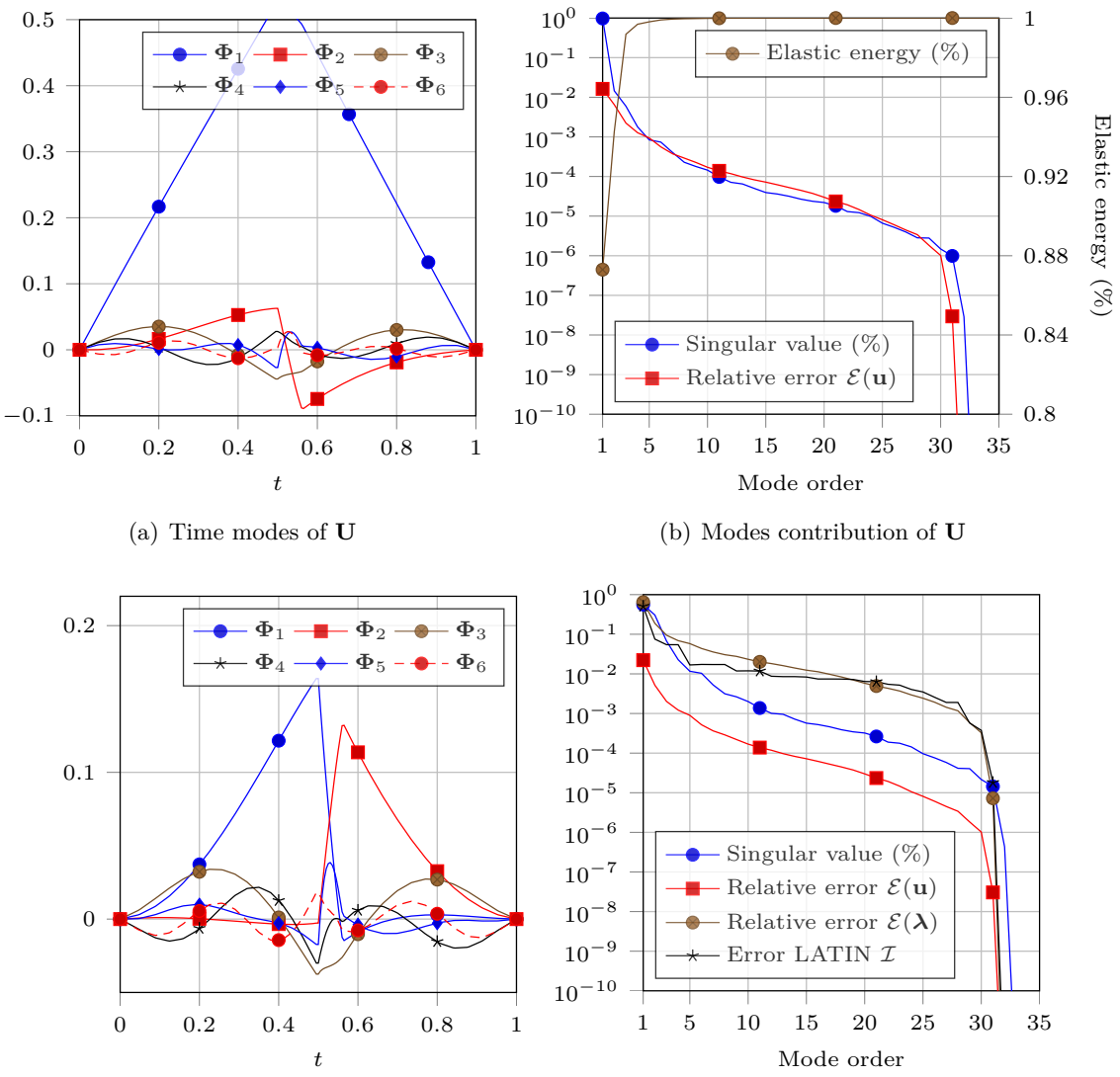


(a) Mode 1 (b) Mode 2 (c) Mode 3 (d) Mode 4 (e) Mode 5 (f) Mode 6 (g) Mode 7 (h) Mode 8

(i) Mode* 1 (j) Mode* 2 (k) Mode* 3 (l) Mode* 4 (m) Mode* 5 (n) Mode* 6 (o) Mode* 7 (p) Mode* 8

Figure 1.9 – SVD space modes of **U** and **U*** (denoted by mode*) scaled by a corresponding factor $0.2\sqrt{\sigma_i}$. The color map refers to the norm of the tensor strain field $\|\boldsymbol{\varepsilon}\| = \sqrt{\boldsymbol{\varepsilon} : \boldsymbol{\varepsilon}}$.

The SVD of **U** provides space modes (see figure 1.9), time modes, singular values amplitudes and energy distribution (see figure 1.10). Remarkably, to rebuilt exactly (up the machine zero point) the displacement field **u**, only 33 SVD modes are needed. To qualify



(a) Time modes of $\mathbf{U}$

(b) Modes contribution of $\mathbf{U}$

(c) Time modes of $\mathbf{U}^*$

(d) Modes contribution of $\mathbf{U}^*$

Figure 1.10 – Modes contribution for both $\mathbf{U}$ and $\mathbf{U}^*$ and SVD time modes scaled by a corresponding factor $0.2\sqrt{\sigma_i}$.

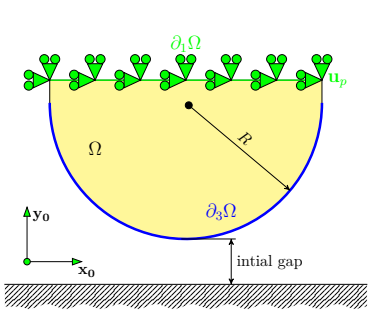
the rebuilt displacement field $\check{\mathbf{u}}$ with a given number K of space-time modes, both relative error according to the criterion 1.14 and the captured elastic energy may be considered. Obviously, the more elevated is K, the more accurate is the approximation $\check{\mathbf{u}}$. For instance, if one takes 5 space-time modes, the rebuilt displacement field $\check{\mathbf{u}}$ captures 99.8% of the integrated elastic energy and its relative error is less than 0.09%. Consequently, the displacement field $\mathbf{u}$ is highly reducible. In practice, 2 or 3 modes could be considered to compress $\mathbf{u}$ efficiently.

The SVD of $\mathbf{U}^*$ provides also space modes (see figure 1.9), time modes with their

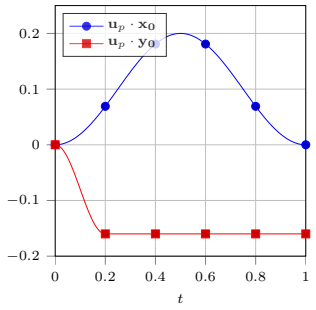
corresponding amplitudes (see figure 1.10). In addition, errors defined by formula (1.40) for both displacement field and contact force field are plotted according to the number of modes considered for their corresponding low rank approximation. One can notice that, thanks to the heritage of homogeneous Dirichlet boundary conditions, each space mode is kinematically admissible to zero. Once again and for the same observation, solution fields $\mathbf{u}^*$ as well as $\boldsymbol{\lambda}$ are highly reducible (even if $\boldsymbol{\lambda}$ is seemingly less compressible than $\mathbf{u}^*$).

1.5.2 Fretting problem

The fretting problem (figure 1.11) consists in an elastic hemicylindrical body contacting a frictional rigid basement. Displacements on the upper boundary are prescribed such that a large contact zone occurs. The scale of the structure and the scale of the contact zone are similar. Initially, the solid is not contacting. From time-step t_1 to t_{40}, it approaches and becomes pressed on the rigid basement. Meanwhile, a tangential displacement is prescribed to activate a global frictional reaction at the contacting zone. Some snapshots of the solution are given in figure 1.12.



(a) Numerical model

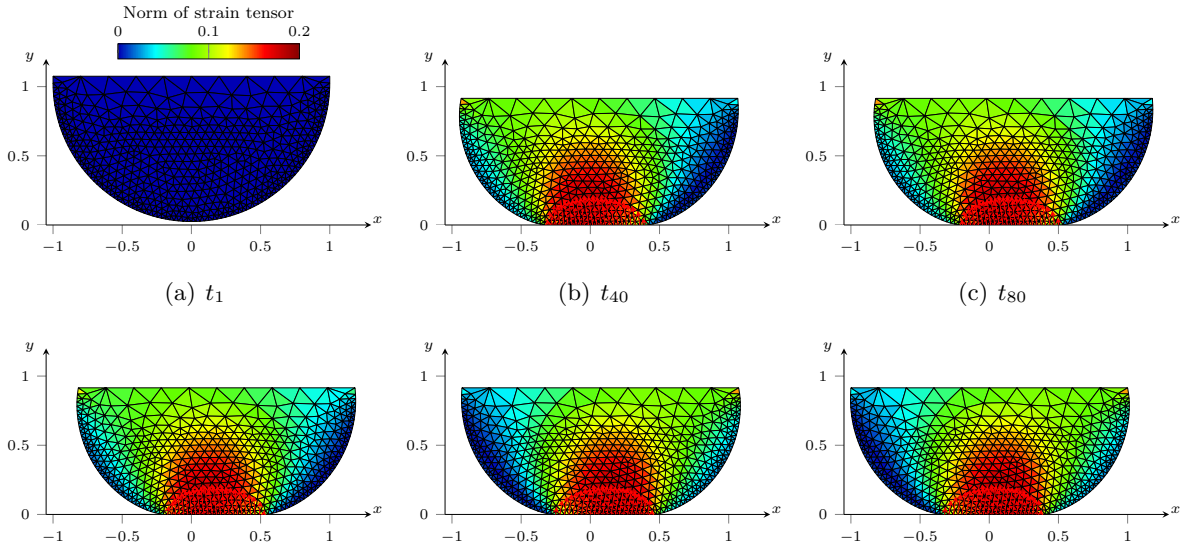


(b) Prescribed displacements

Property		Value
Young modulus	E	3000 Pa
Poisson ratio	ν	0.3
Frictional coefficient	μ	0.15
Time interval	$[0, T]$	$[0, 1]$
Time stepping	Δt	$5 \cdot 10^{-3}$ s
Degrees of freedom	n	1144
Contacting nodes	n_C	101

(c) Properties

Figure 1.11 – Hemicylindrical body contacting a rigid basement.

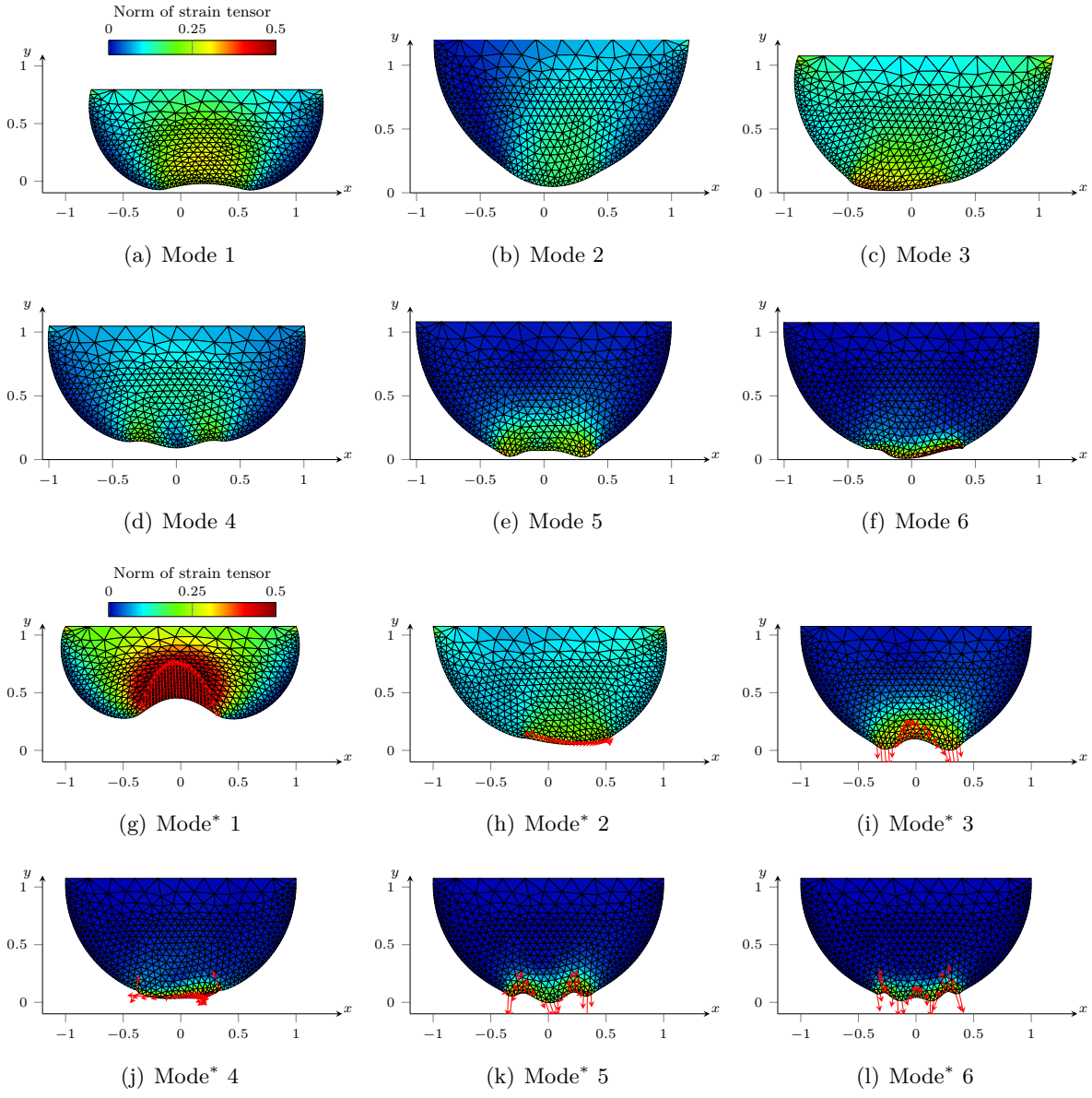


(a) t_1 (b) t_{40} (c) t_{80}

(d) t_{120} (e) t_{160} (f) t_{200}

Figure 1.12 – Solution $\mathbf{s}$: displacement and contact force fields. The color map refers to the norm of the strain tensor field $\|\boldsymbol{\varepsilon}\| = \sqrt{\boldsymbol{\varepsilon} : \boldsymbol{\varepsilon}}$ and red arrows correspond to nodal contact force.

The SVD of both $\mathbf{U}$ and $\mathbf{U}^*$ provide space modes (figure 1.13) and time modes (figure



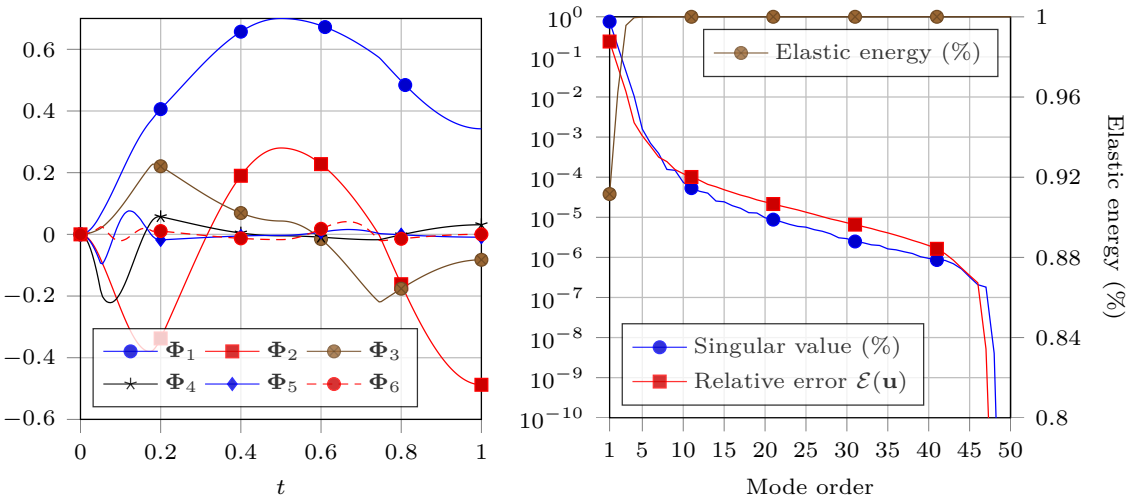
(a) Mode 1 (b) Mode 2 (c) Mode 3

(d) Mode 4 (e) Mode 5 (f) Mode 6

(g) Mode* 1 (h) Mode* 2 (i) Mode* 3

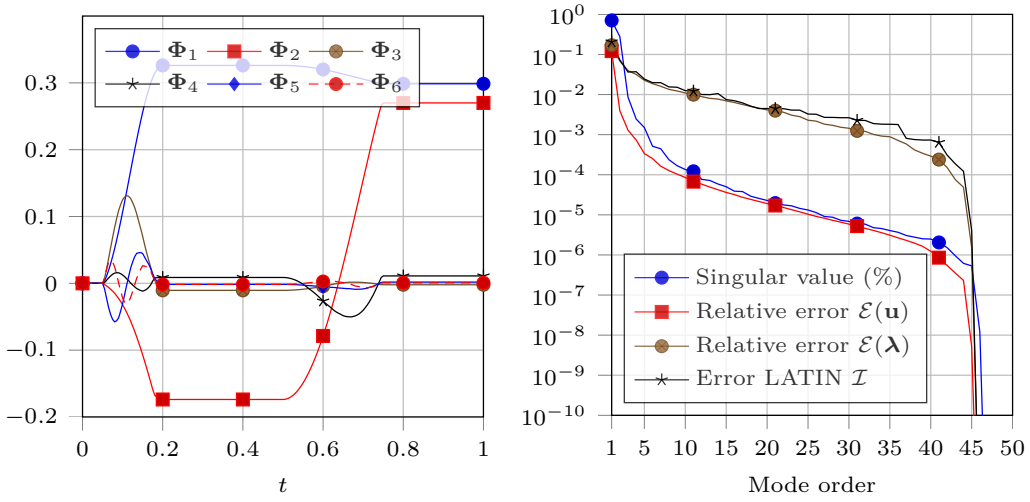
(j) Mode* 4 (k) Mode* 5 (l) Mode* 6

Figure 1.13 – SVD space modes of $\mathbf{U}$ and $\mathbf{U}^*$ (denoted by mode*) scaled by a corresponding factor $\sqrt{\sigma_i}$. The color map refers to the norm of the tensor strain field $\|\boldsymbol{\varepsilon}\| = \sqrt{\boldsymbol{\varepsilon} : \boldsymbol{\varepsilon}}$.



(a) Time modes of $\mathbf{U}$

(b) Modes contribution of $\mathbf{U}$



(c) Time modes of $\mathbf{U}^*$

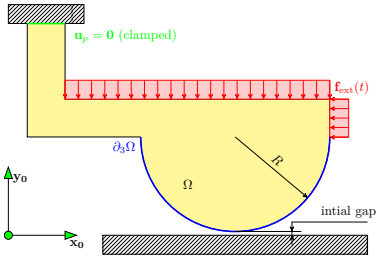
(d) Modes contribution of $\mathbf{U}^*$

Figure 1.14 – Modes contribution for both $\mathbf{U}$ and $\mathbf{U}^*$ and SVD time modes scaled by a corresponding factor $\sqrt{\sigma_i}$.

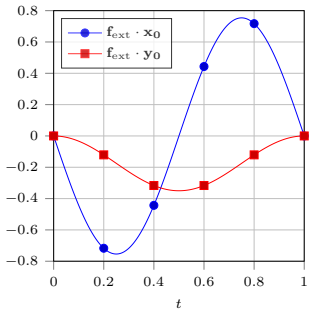
1.14). Once again, both displacement fields $\mathbf{u}$ and $\mathbf{u}^*$ are reducible and less than 5 modes are required to rebuilt accurately the whole space-time solution.

1.5.3 Cantilevered indenter

This problem (figure 1.15) consists in an elastic cantilevered hemicylindrical indenter contacting a frictional rigid basement. External loads are designed in order to get different status at the contacting nodes during the studied time interval (no contact, sliding, sticking). The size of the effective contact area is smaller than the scale of the structure. Some snapshots of the solution are given in figure 1.16.



(a) Numerical model



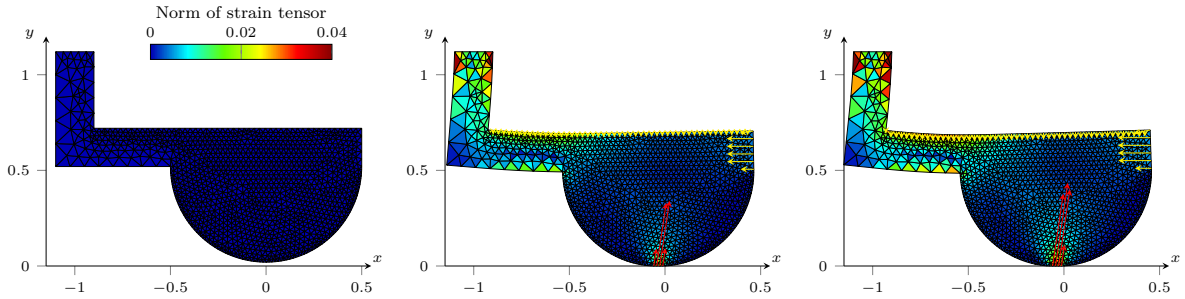
(b) Boundaries

Property		Value
Young modulus	E	3000 Pa
Poisson ratio	ν	0.3
Frictional coefficient	μ	0.15
Time interval	$[0, T]$	$[0, 1]$
Time stepping	Δt	$5 \cdot 10^{-3}$ s
Degrees of freedom	n	2564
Contacting nodes	n_C	101

(c) Properties

Figure 1.15 – Cantilevered contacting a rigid basement.

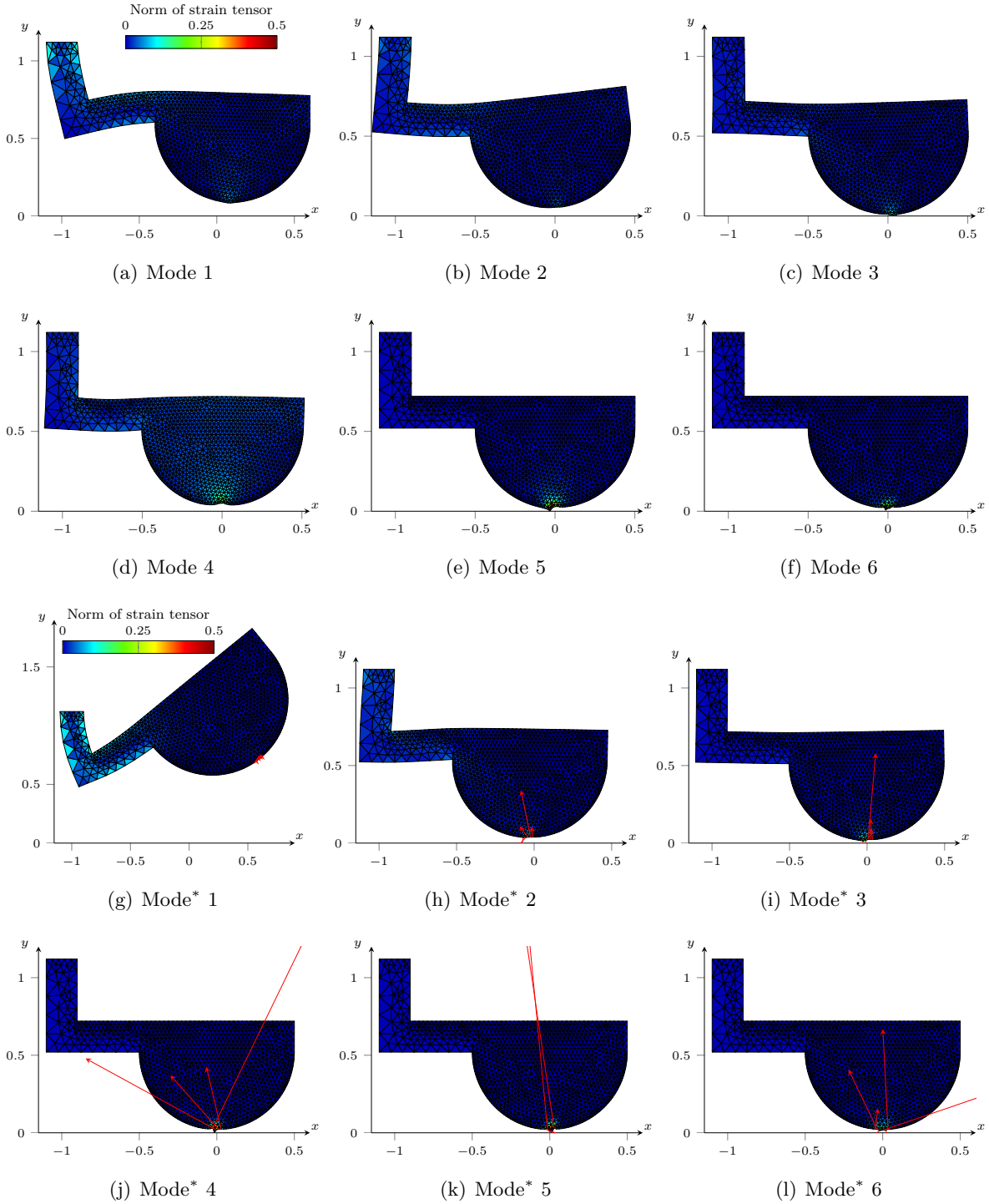
From figures 1.17 and 1.18, same conclusions as for the previous cases can be drawn. Status switching seems to not affect reducibility of the solution. For this problem, the length scale of the global structure is more separated from the one of the contact area, and this is depicted in the succession of spatial modes.



(a) t_1 (b) t_{40} (c) t_{80}

(d) t_{120} (e) t_{160} (f) t_{200}

Figure 1.16 – Solution $\mathbf{s}$: displacement and contact force fields. The color map refers to the norm of the strain tensor field $\|\boldsymbol{\varepsilon}\| = \sqrt{\boldsymbol{\varepsilon} : \boldsymbol{\varepsilon}}$, red arrows correspond to nodal contact force and yellow arrows correspond to the external load.



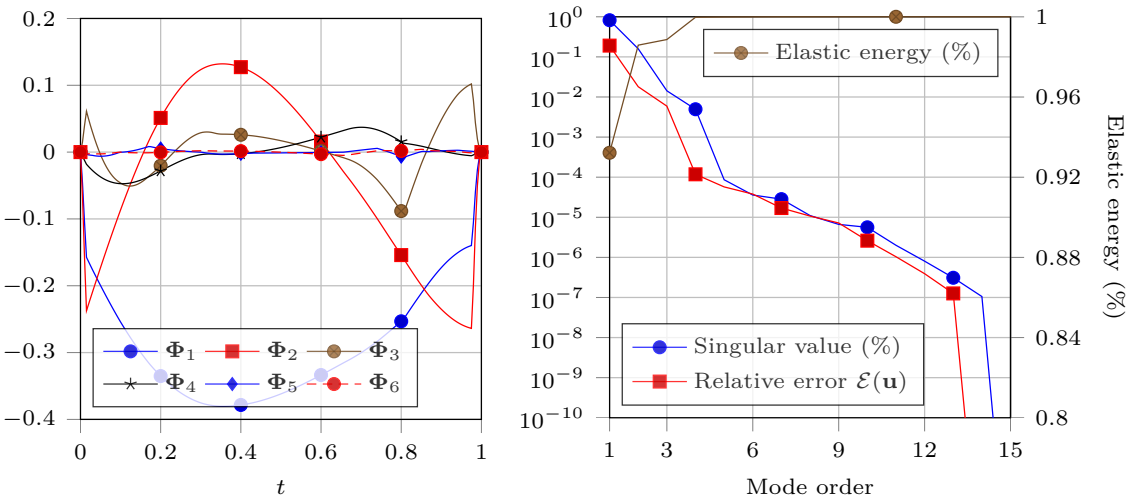
(a) Mode 1 (b) Mode 2 (c) Mode 3

(d) Mode 4 (e) Mode 5 (f) Mode 6

(g) Mode* 1 (h) Mode* 2 (i) Mode* 3

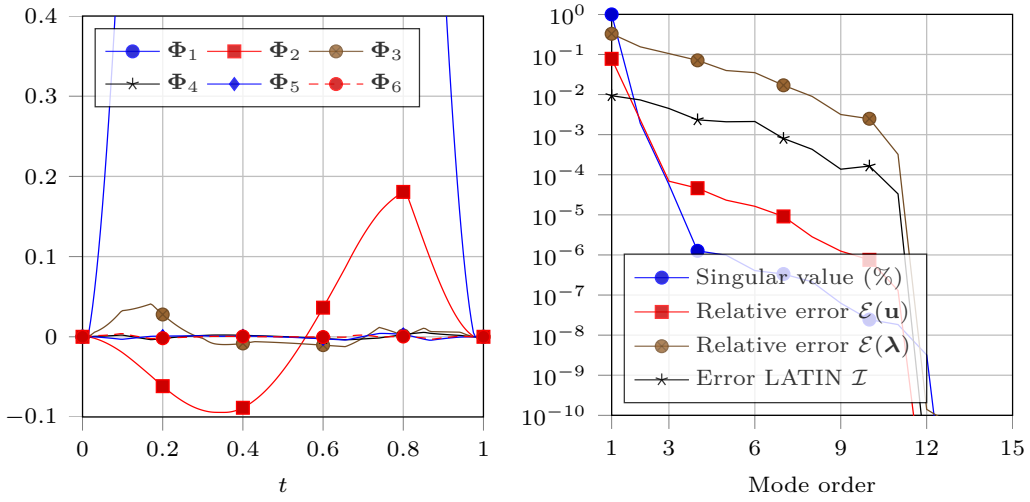
(j) Mode* 4 (k) Mode* 5 (l) Mode* 6

Figure 1.17 – SVD space modes of $\mathbf{U}$ and $\mathbf{U}^*$ (denoted by mode*) scaled by a corresponding factor $\sqrt{\sigma_i}$. The color map refers to the norm of the tensor strain field $\|\boldsymbol{\varepsilon}\| = \sqrt{\boldsymbol{\varepsilon} : \boldsymbol{\varepsilon}}$.



(a) Time modes of $\mathbf{U}$

(b) Modes contribution of $\mathbf{U}$



(c) Time modes of $\mathbf{U}^*$

(d) Modes contribution of $\mathbf{U}^*$

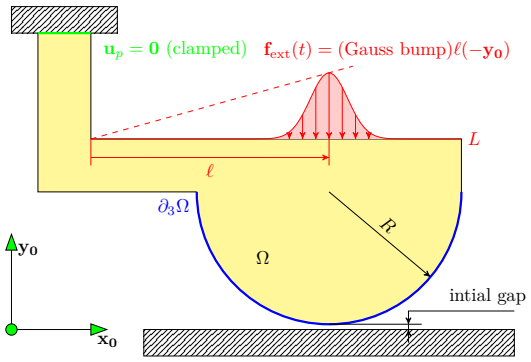
Figure 1.18 – SVD time modes of $\mathbf{U}$ scaled by a corresponding factor $\sqrt{\sigma_i}$, singular value amplitudes and elastic energy distribution.

1.5.4 Cantilevered indenter with a mobile charge

This problem (figure 1.19) is the same as the previous one excepting for the external loads. A mobile Gaussian-lobe load moving along the upper boundary of the body is designed with an increasing amplitude. It is well-known that this kind of solicitation is not efficiently described by a finite set of separated function (1.13). A soft contact occur at the bottom of the hemicylindrical part. In figure 1.19, a space variable ℓ localizing the position of the lobe is defined. The evolution of ℓ according the time variable is prescribed as follows:

$$\forall \ell \in [0, L] : \begin{cases} t \in [0, \frac{T}{4}] : & \ell = 4L\frac{t}{T} \\ t \in [\frac{T}{4}, \frac{T}{2}] : & \ell = 4L(1 - \frac{t}{T}) \\ t \in [\frac{T}{2}, \frac{3T}{4}] : & \ell = 4L\frac{t}{T} \\ t \in [\frac{3T}{4}, T] : & \ell = 4L(1 - \frac{t}{T}) \end{cases} \tag{1.41}$$

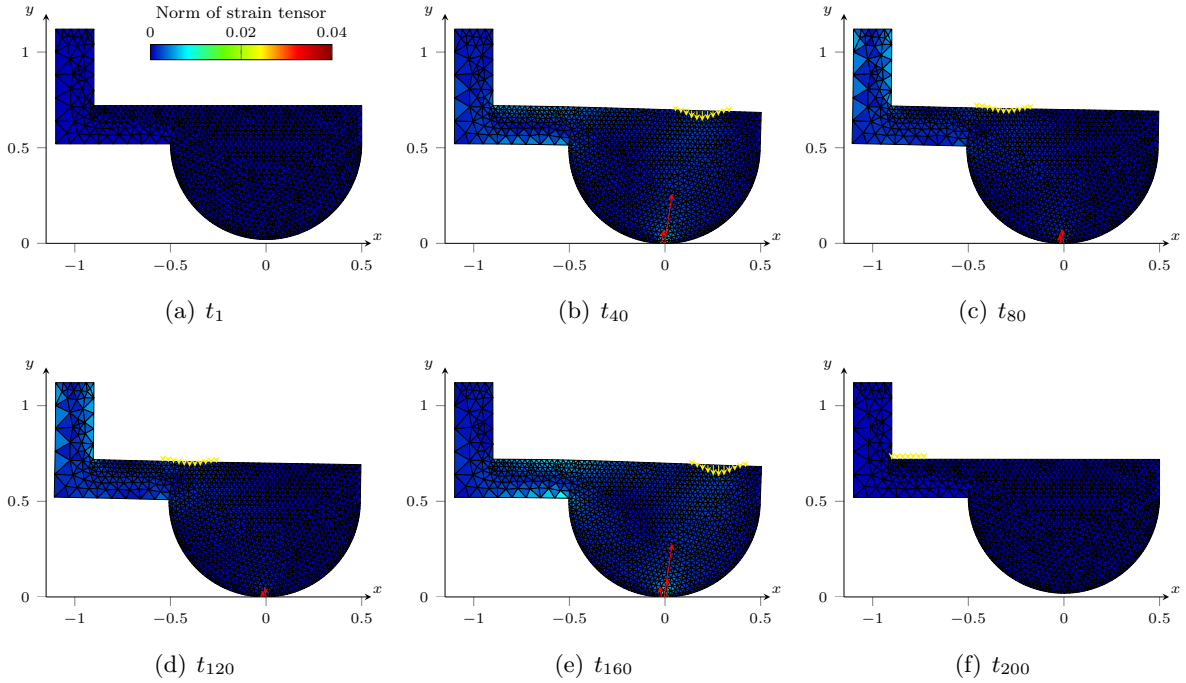
All in all, the mobile charge moves twice, from an abscissa 0 to L and then comes back to 0.



(a) Numerical model

Figure 1.19 – Cantilevered indenter with a mobile charge contacting a rigid basement (properties are the same as the cantilevered indenter). The mobile charge sweeps the upper boundary twice.

With the mobile charge, the displacement field $\mathbf{u}$ is clearly less compressible relatively to the previous case (see figure 1.22b to compare with 1.18b). More modes are required to rebuild the snapshots $\mathbf{U}$ (to capture 99% of the elastic energy, more than 10 modes are required instead of 4 for the previous case). One can remark that the displacement field $\mathbf{u}^*$ is highly reducible. Indeed, $\mathbf{u}^*$ is obtained from $\mathbf{u}$ by subtracting the displacement field associated to the external load. As a consequence, the poor compressible part of $\mathbf{u}$ (*i.e.* induced by the mobile charge) vanishes and the remaining $\mathbf{u}^*$ is in this case highly compressible. All exemplified displacement fields $\mathbf{u}^*$ show a great compressible property. Nonetheless, for large sliding motions involving a moving effective contact zone, the same



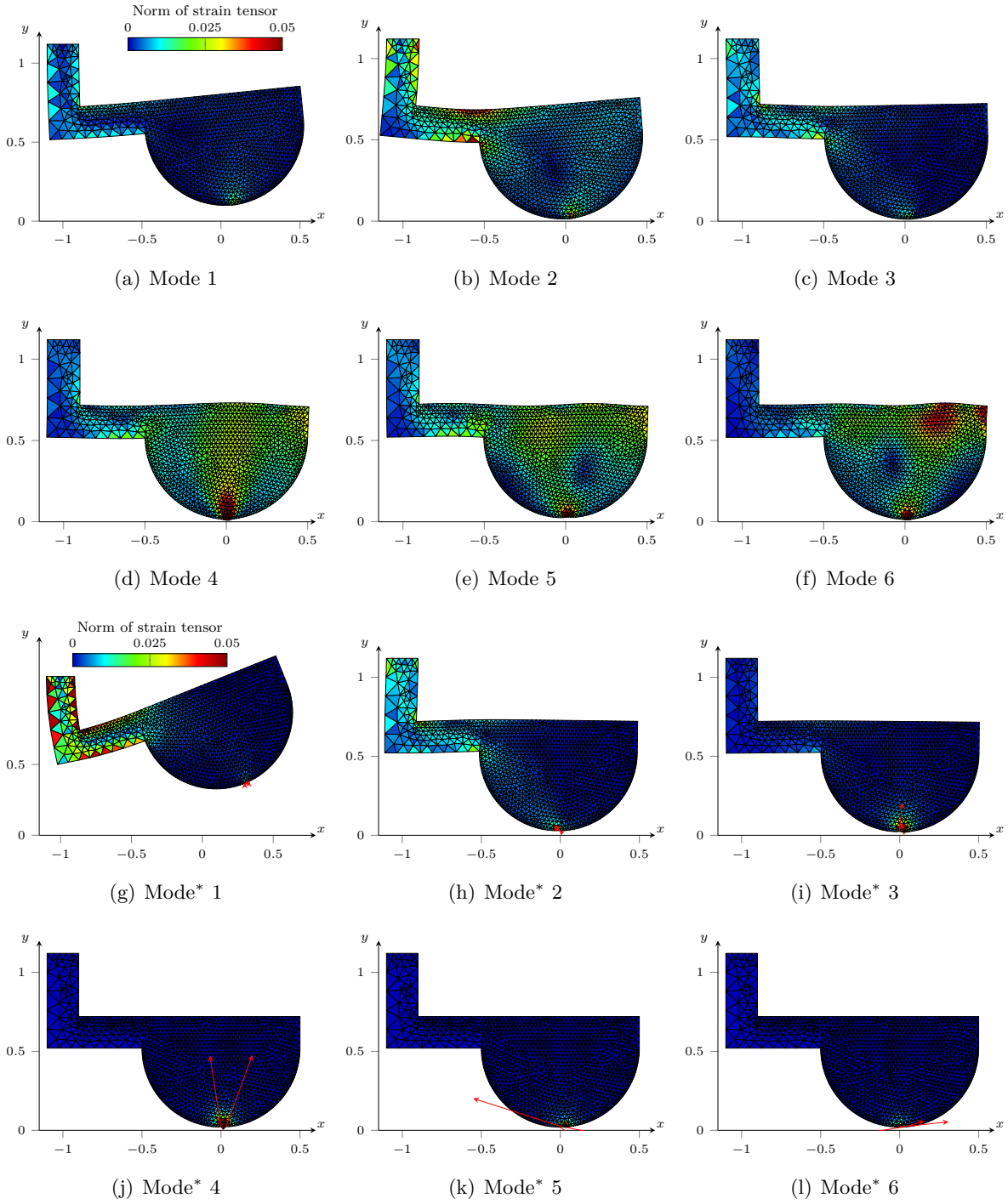
(a) t_1 (b) t_{40} (c) t_{80}

(d) t_{120} (e) t_{160} (f) t_{200}

Figure 1.20 – Solution $\mathbf{s}$: displacement and contact force fields. The color map refers to the norm of the strain tensor field $\|\boldsymbol{\varepsilon}\| = \sqrt{\boldsymbol{\varepsilon} : \boldsymbol{\varepsilon}}$, red arrows correspond to nodal contact force and yellow arrows correspond to the external load.

behavior as described above occurs. Then, displacement field $\mathbf{u}^*$ would become poorly compressible.

Finally, one can also notice the periodicity of time modes due to the periodicity of the mobile charge (two sweeps).



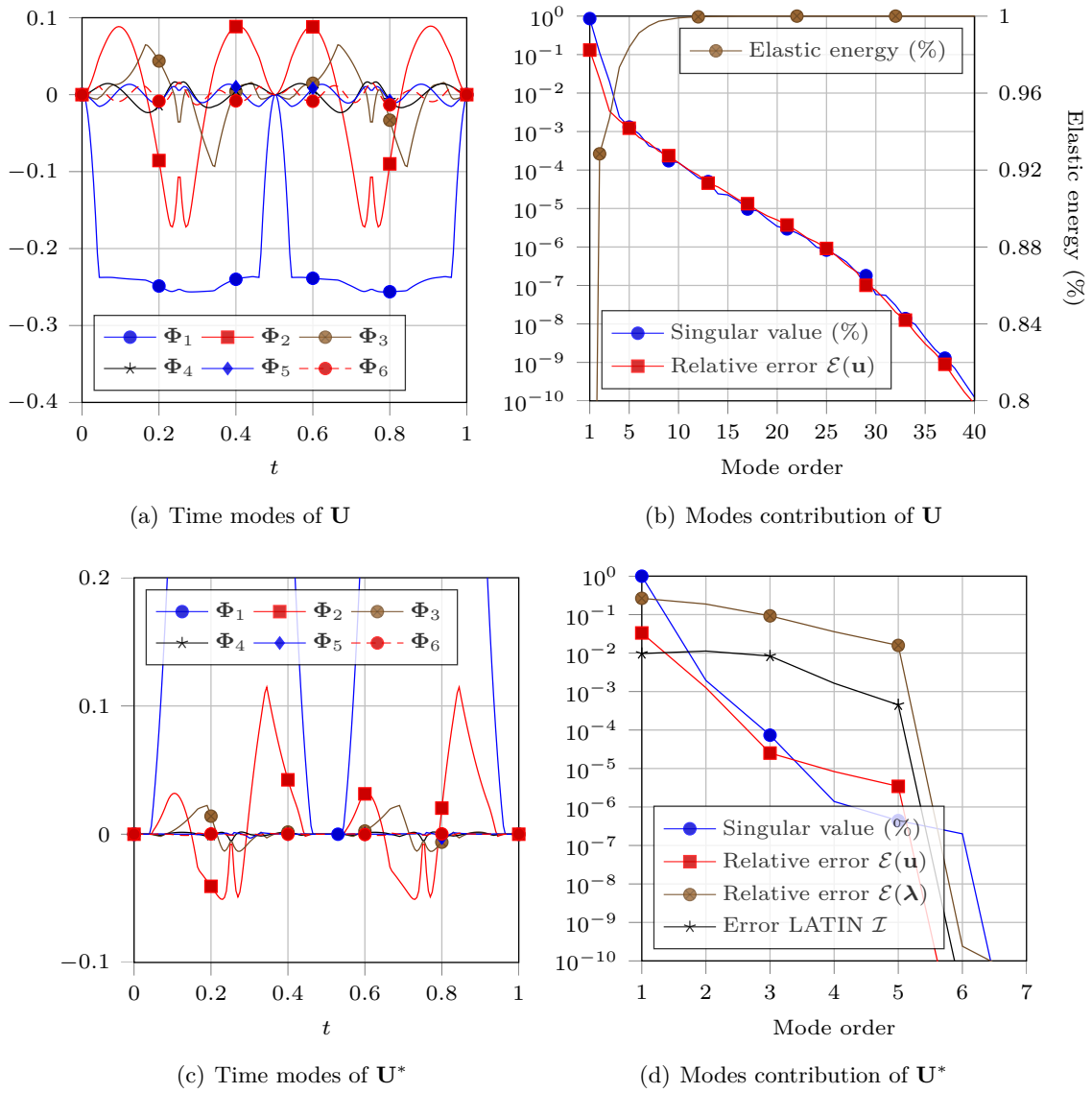
(a) Mode 1 (b) Mode 2 (c) Mode 3

(d) Mode 4 (e) Mode 5 (f) Mode 6

(g) Mode* 1 (h) Mode* 2 (i) Mode* 3

(j) Mode* 4 (k) Mode* 5 (l) Mode* 6

Figure 1.21 – SVD space modes of $\mathbf{U}$ and $\mathbf{U}^*$ (denoted by mode*) scaled by a corresponding factor $\sqrt{\sigma_i}$. The color map refers to the norm of the tensor strain field $\|\boldsymbol{\varepsilon}\| = \sqrt{\boldsymbol{\varepsilon} : \boldsymbol{\varepsilon}}$.



(a) Time modes of $\mathbf{U}$

(b) Modes contribution of $\mathbf{U}$

(c) Time modes of $\mathbf{U}^*$

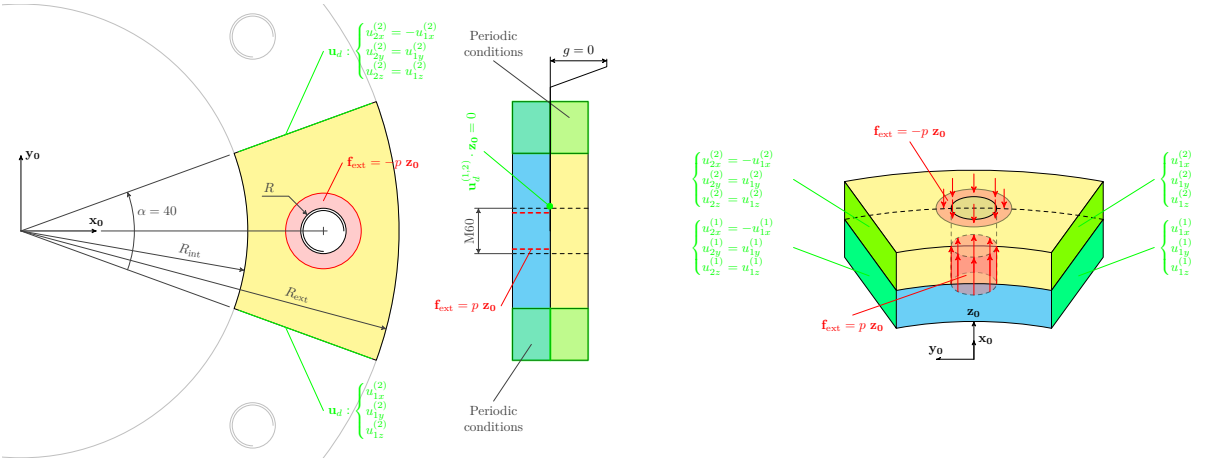
(d) Modes contribution of $\mathbf{U}^*$

Figure 1.22 – SVD time modes of $\mathbf{U}$ scaled by a corresponding factor $\sqrt{\sigma_i}$, singular value amplitudes and elastic energy distribution.

1.5.5 Three dimensional multibody problem

The following example consists in an assembly of two bolted annular rings (see figure 1.23). Instead of dealing with full rings, only a 40° sector is considered and periodic boundary conditions are prescribed for the displacement field. Boundaries are designed such that, first a screwing force pinches the two rings, then, small displacements occur to activate friction. The solution is assumed to be known and is plotted for some time steps on figures 1.25 and 1.26.

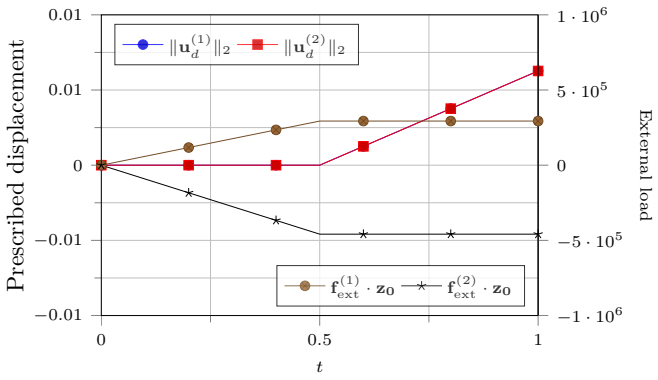
Unlike previous examples, multibody systems require a special attention as far as the discretization and the contact detection are concerned. For the sake of simplicity, we use a node-to-node discretization [FZ75] with conforming meshes for involved bodies. Then, contact condition are easily written at each pair of nodes. Such a discretization is very simple to implement and to handle numerically. Its great drawbacks are the need of conforming meshes, and the fact that only small deformation and small slip can be handled. On the one hand, to address those limitations, other methods such as node-to-segment [HTS+76, ZDL09] or segment-to-segment [WS85, PL04] were developed. Nonetheless, one has to be advised that all of these methods are not ensured to pass the Taylor's patch test [TP91, EAB01]. On the other hand, mortar [BMP93, LM97, BHL98, ML00, Woh00, Hil00] and Nitsche [BHS03, WZ08, FHW04] methods seem to be the most efficient to take into account large deformations and slip between bodies non-conformingly meshed. These methods pass the Taylor's test but the price to pay is their highly complicated implementation (namely for 3D cases).



(a) Normalized draft

(b) Perspective

Figure 1.23 – Model of the bolted annular rings.

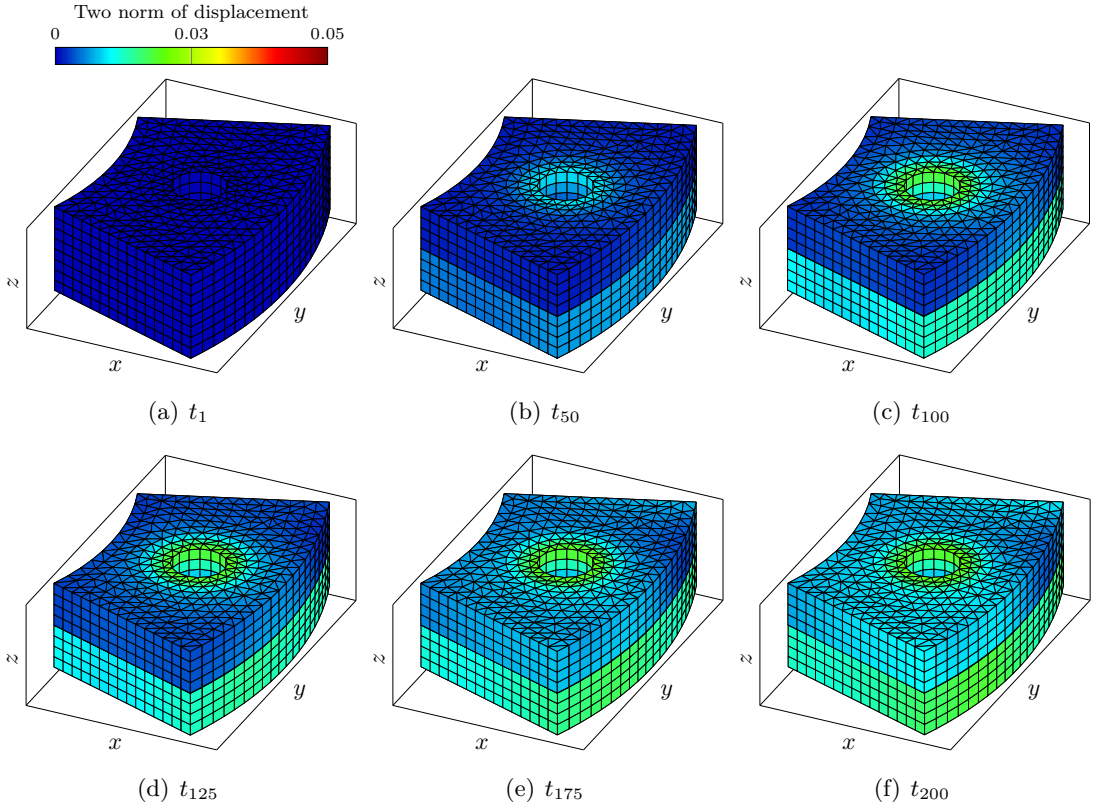


(a) Boundary conditions (norms of prescribed displacements are overlapped)

Property		Value
Young modulus	E	1 GPa
Poisson ratio	ν	0.3
Frictional coefficient	μ	0.5
Time interval	$[0, T]$	$[0, 1]$
Time stepping	Δt	$5 \cdot 10^{-3}$ s
Degrees of freedom	n	2×4845
Contacting nodes	n_C	2×291

(b) Properties

Figure 1.24 – Bolted annular rings.

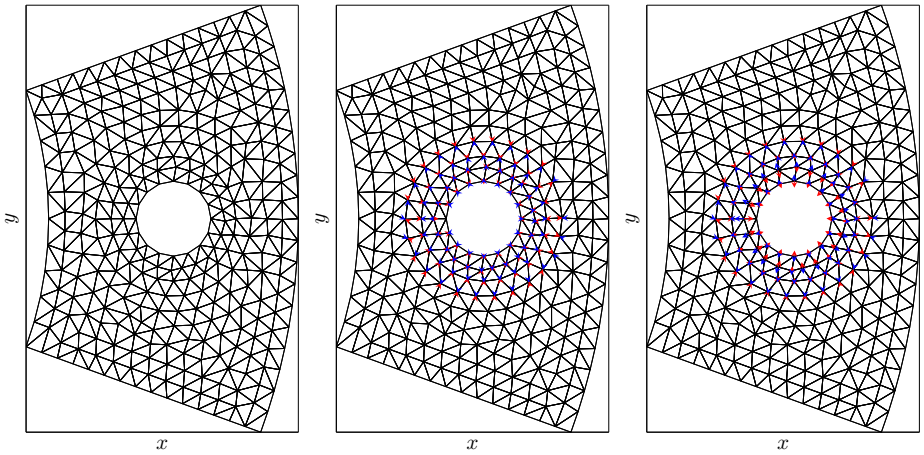


(a) t_1 (b) t_{50} (c) t_{100}

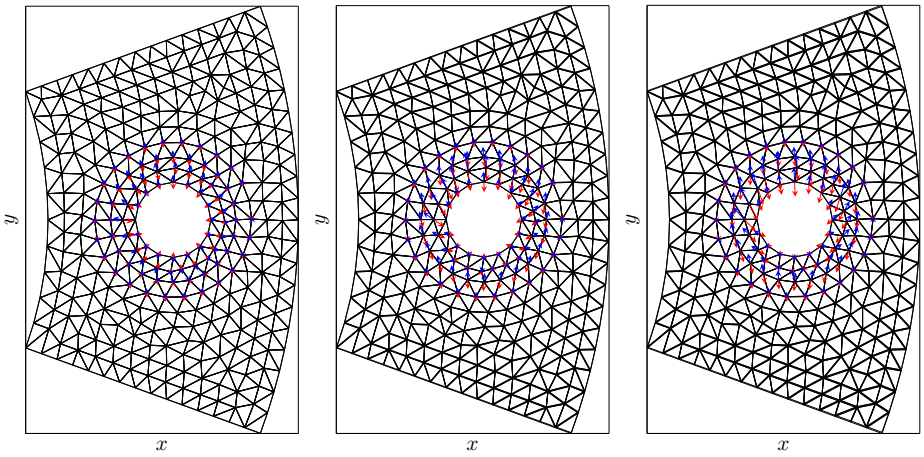
(d) t_{125} (e) t_{175} (f) t_{200}

Figure 1.25 – Solution: displacement field. The color map refers to the norm of the displacement field.

Solution of this problem includes the solution for the first body $\mathbf{s}^{(1)}$ and the solution for the second body $\mathbf{s}^{(2)}$. Displacement fields are concatenated into the same snapshot matrix



(a) t_1 (b) t_{50} (c) t_{100}



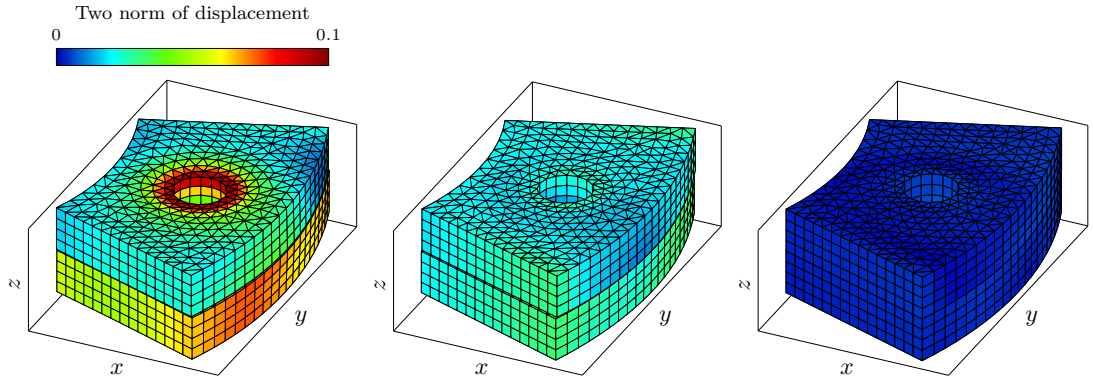
(d) t_{125} (e) t_{175} (f) t_{200}

Figure 1.26 – Contact force field. Red and blue arrows correspond respectively to the first (lower) and second (upper) body nodal contact force.

from which the SVD is performed. But, one may carry out following SVD analyses for each individual body. The SVD of $\mathbf{U}$ provides space modes (see figure 1.27), time modes, singular values amplitudes and energy distribution (see figure 1.29). As the size of the problem is larger than previously, less than roughly 90 space-time modes are required to rebuilt exactly the solution. Remarkably, accounting 2 space-time modes leads to a low rank approximation $\check{\mathbf{u}}$ with a relative error of 0.8% relatively to $\mathbf{u}$ capturing 99.8% of the integrated elastic energy. Again, field $\mathbf{u}$ is highly reducible.

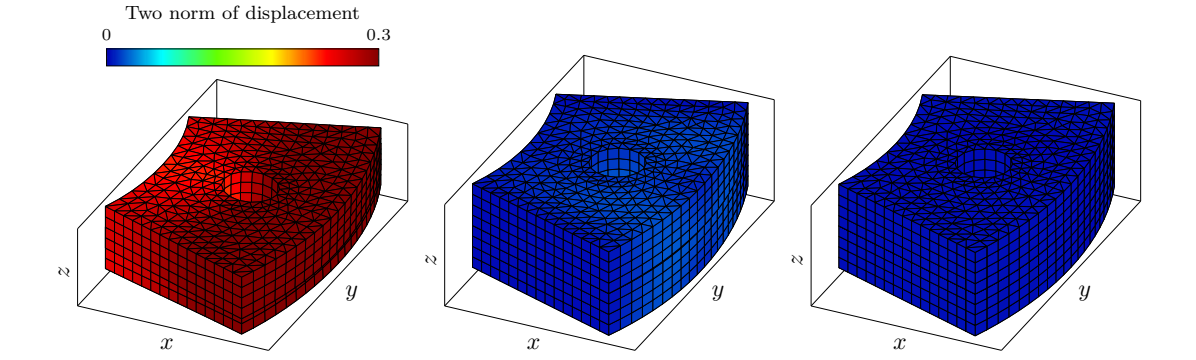
The SVD of $\mathbf{U}^*$ provides space modes (see figure 1.28), time modes with their corresponding amplitudes (see figure 1.29). In addition, errors defined by formula (1.40) for both displacement field and contact force field are plotted according to the number of modes considered for their correspond low rank approximation. Again and for same

observation, solution fields $\mathbf{u}^*$ as well as $\boldsymbol{\lambda}$ are highly reducible even if $\boldsymbol{\lambda}$ is seemingly less compressible than $\mathbf{u}^*$.

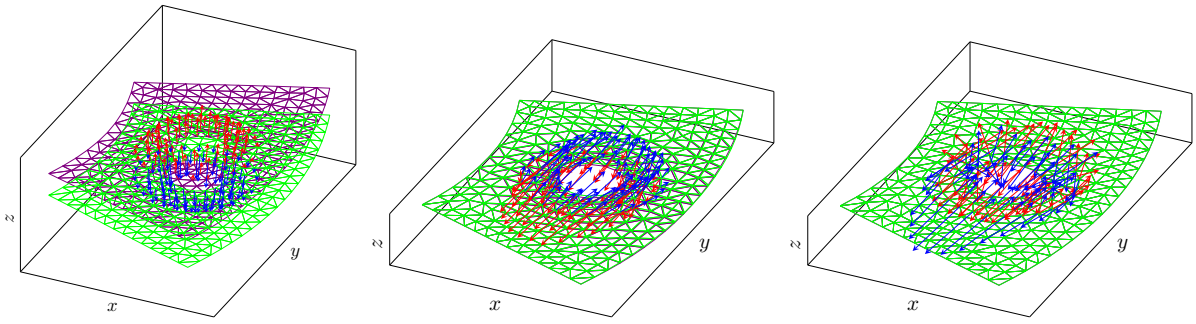


(a) Mode 1 (b) Mode 2 (c) Mode 3

Figure 1.27 – SVD space modes of $\mathbf{U}$ scaled by a corresponding factor $\sqrt{\sigma_i}$. The color map refers to the norm of the displacement field.

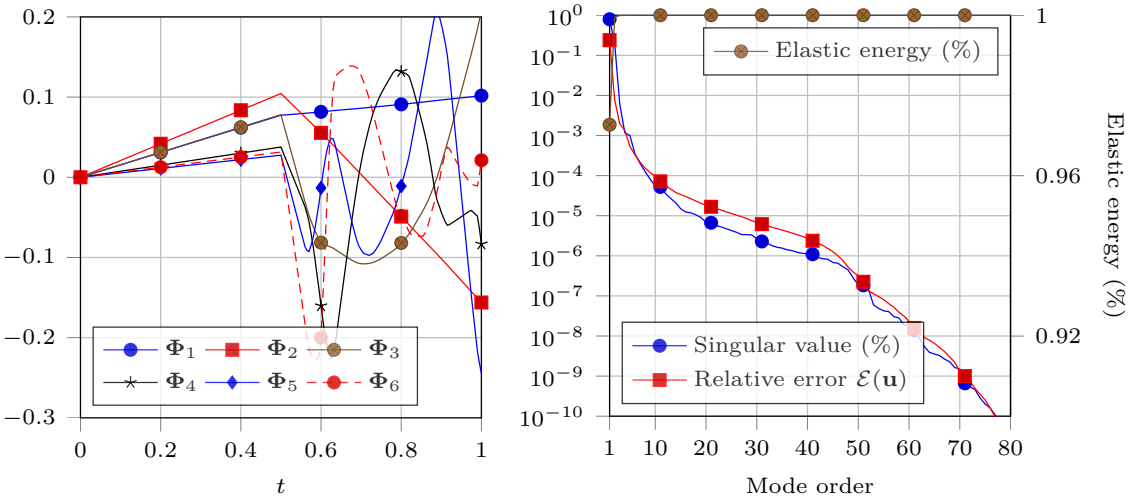


(a) Displacement mode 1 (b) Displacement mode 2 (c) Displacement mode 3



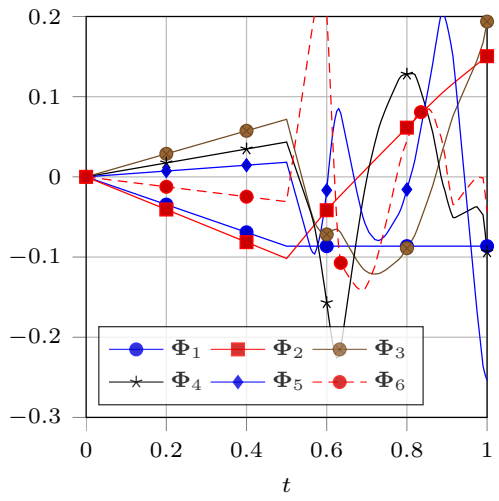
(d) Contact force mode 1 (e) Contact force mode 2 (f) Contact force mode 3

Figure 1.28 – SVD space modes of $\mathbf{U}^*$ scaled by a corresponding factor $\sqrt{\sigma_i}$ (the color map refers to the norm of the displacement field). Red and blue arrows correspond respectively to the first (lower) and second (upper) body nodal contact force.

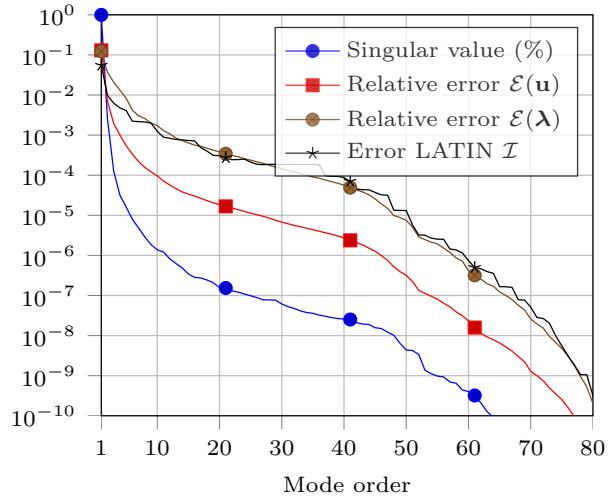


(a) Time modes of $\mathbf{U}$

(b) Modes contribution of $\mathbf{U}$



(c) Time modes of $\mathbf{U}$



(d) Modes contribution of $\mathbf{U}^*$

Figure 1.29 – Modes contribution for both $\mathbf{U}$ and $\mathbf{U}^*$ and SVD time modes scaled by a corresponding factor $\sqrt{\sigma_i}$.

1.5.6 Scale separability observation

From all of these previous SVD analyses, great compressible properties were exemplified. Indeed for the presented cases, a few modes are needed to provide an accurate approximation for fields of interest. Last but not least in addition to the reducibility, a scale separability phenomenon is enlightened. For all problems, two scales can be distinguished at least: a global / structural scale and a scale localized at the contacting interface. For the cantilevered indenter with a mobile charge a third scale related to the moving boundary condition can be also identified.

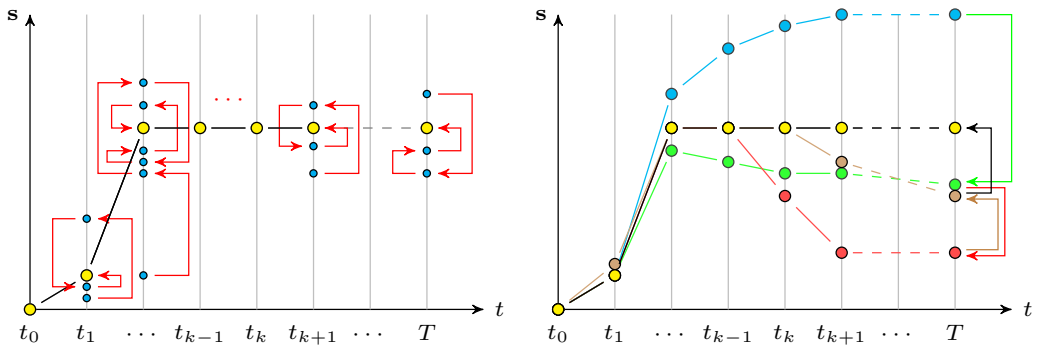
The first SVD modes (most energetic ones) include global trends of the analyzed field. Whereas higher order modes bring local corrections enhancing precision at contact interface. Time modes are also affected by the scale separability. First time modes are associated to global space mode to depict structural scale displacements ($\sim$ low frequency) whereas higher order ones correspond to local space modes bringing correction at the contacting interface (higher frequency). Such observations were also remarked in [GDGR14].

All of these remarks show that a frictional contact solution and adopting a global space-time approach is reducible and has remarkable different scales. In other words, a relevant and a multiscale reduced basis for the solution can be computed with for instance a sample of SVD space modes.

1.6 The large time increment (LATIN) method

1.6.1 Space-time global approach

Incremental vs. non-incremental



(a) Incremental approach (b) Non-incremental approach

Figure 1.30 – Comparison between an incremental and a non-incremental iterative solver.

To tackle a time dependent non-linear problem, several non-linear iterative solvers are

suited. Generally, these can be distinguished into two families: incremental solvers and non-incremental ones. The major difference between them lies on the manner the time dimension is handled.

An incremental solver consists in making iterations to solve the non-linear problem for a given time step t_k (*e.g.* until the internal balance equation is reached). Once the convergence criterion is ensured (*e.g.* norm on the equilibrium residual), thesolution found at t_k is stored and the same process is repeated for t_{k+1}, etc, up to T. This is the most widespread procedure in commercial codes (*e.g.* Newton-Raphson, quasi-Newton, Newton modified methods).

On the contrary, non-incremental solvers iterate on the space domain but also on the whole time interval (all time steps are swept at each iteration). Each non-incremental iteration ends on an space-time approximation of the solution. The iterating process is stopped once a global space-time convergence criterion (*e.g.* based on constitutive laws) is reached. This is one of the major point of the large time increment (LATIN) method well-known for its ability to solve general non-linear problems. Nevertheless, memory requirements for such a method are consequent because space-time fields have to be stored and processed. That's why, the LATIN method was proposed with the radial approximation consisting in searching the solution into a low rank approximation presaging nowadays PGD method [Lad99].

Toward reduction methods for frictional contact

Generally speaking, an incremental approach combined with an a priori reduction method lacks in efficiency. Indeed, the time dependent non-linear problem is attempted to be solved in a reduced basis at a fixed time step. The reduced basis can be extended using enriching strategies as in [Ryc05, RCCA06] until the solution fulfills a convergence criterion. By doing so, obtained modes may not be the most significant in regards to the converged solution (*i.e.* lack in identifying the different scales of the problem). Particularly for frictional problems, numerous enrichments can be required to capture status switches occurring between time steps in order to fulfill a convergence criterion. Then, benefits of dimensionality reduction are lost. That's why, reduction methods seem to be inappropriate and cumbersome for such problems.

Nonetheless, space-time approaches with a non-incremental method can circumvent this difficulty. Indeed, exploring the whole time interval at each iteration enables to identity appropriate modes (*i.e.* capture scales of the problem). As a consequence, the problem is expected to be solved into a more and more relevant subspace. Indeed, computed modes are globally (in the space-time sense) suited to the full space-time problem.

1.6.2 Formulation of the LATIN method for frictional contact

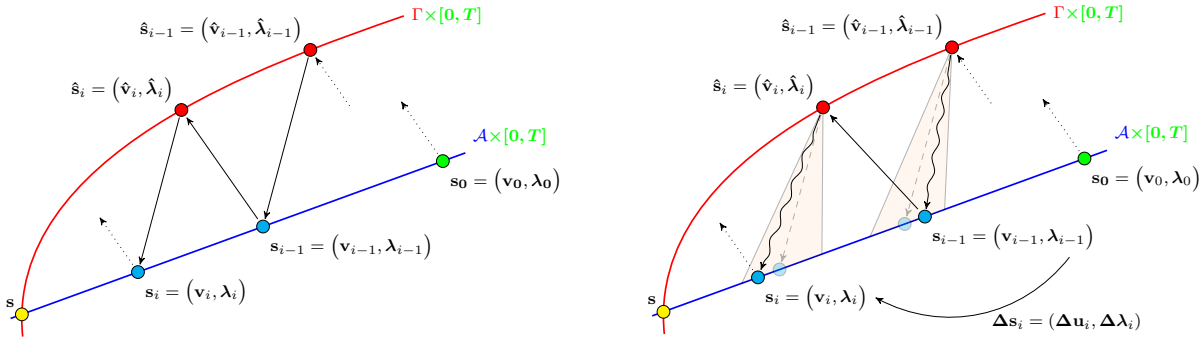
In this dissertation, one of the major ingredients of proposed strategy is the (LATIN) method introduced in [Lad99]. Hereinafter, its formulation for frictional contact problems is given. This method closed to augmented Lagrangian methods is well-known for its ability [HT96] to solve difficult non-linear and time-dependent large problems with a global time-space approach (non-linear material [BBL90, CL93, RNB13], contact problems [BLPR97, CCDL97, CCL99, GDGR14], large displacement [BLPR97], transient dynamics [LLB00, LBL02, OBG10], fracture mechanics [DMB01, RBDG07, PBG10, TGBNT12] ...). Originally, the non-incremental LATIN method was proposed as a commitment of three principles:

(P1) **Separation of the linear and non-linear behaviors**. $\mathcal{A}$ denotes the set of solutions $\mathbf{s} = (\mathbf{u}, \boldsymbol{\lambda})$ satisfying linear constitutive law, kinematic admissibility and static admissibility. These are defined on the whole space-time domain $\Omega \times [0,\ T]$. $\boldsymbol{\Gamma}$ denotes the set of solutions $\hat{\mathbf{s}} = (\hat{\mathbf{v}}, \hat{\boldsymbol{\lambda}})$ verifying frictional contact conditions and are defined locally at the contacting interface and on the whole time interval $\partial_3\Omega \times [0, T]$. The solution of the problem is $\mathbf{s} \in \mathcal{A} \cap \boldsymbol{\Gamma}$.

(P2) **A two-staged iterative algorithm**. The solution of the problem is searched with the construction of two sequences of approximations belonging alternatively to $\mathcal{A}$ and $\boldsymbol{\Gamma}$. At the i^{th} iteration, the local stage consists in finding $\hat{\mathbf{s}}_i = (\hat{\mathbf{v}}_i, \hat{\boldsymbol{\lambda}}_i) \in \boldsymbol{\Gamma}$ with a search direction $(\hat{\mathbf{s}}_i - \mathbf{s}_{i-1}) = (\hat{\mathbf{v}}_i - \mathbf{v}_{i-1}, \hat{\boldsymbol{\lambda}}_i - \boldsymbol{\lambda}_{i-1}) \in \mathbf{E}^+$. Note that $\mathbf{s}_{i-1} = (\hat{\mathbf{v}}_{i-1}, \hat{\boldsymbol{\lambda}}_{i-1})$ is known from the previous iteration. Then, the global stage consists in finding $\mathbf{s}_i = (\mathbf{v}_i, \boldsymbol{\lambda}_i) \in \mathcal{A}$ with another search direction $(\mathbf{s}_i - \hat{\mathbf{s}}_i) = (\mathbf{v}_i - \hat{\mathbf{v}}_i, \boldsymbol{\lambda}_i - \hat{\boldsymbol{\lambda}}_i) \in \mathbf{E}^-$. Note that $\hat{\mathbf{s}}_i = (\hat{\mathbf{v}}_i,\ \hat{\boldsymbol{\lambda}}_i)$ is known from the previous local stage.

(P3) **Radial approximation or space-time separation**. Unknown fields are represented as a sum of products between a space function and a time function to limit memory usage. This supplementary constraint makes the problem over-determined. In order to respect admissibility conditions stated (P1), the search direction equation $\mathbf{E}^-$ is verified in a weak sense.

These principles are illustrated on figure 1.31. For certain cases and for a sake of simplicity, the LATIN method can be formulated without the space-time separation (P3). In this case several similarities can be stated with augmented Lagrangian methods [ADR06]. All in all, the LATIN method for frictional contact problems consists in global / local strategy whose global stage does not require matrix re-factorization (stiffness operator remains constant, symmetric and definite positive through LATIN iterations) and local stage is explicit (no iterations are required to handle the non-linear behavior at the contacting boundary). As a consequence, comparisons between LATIN and Newton solvers is not an easy task as the number of iterations is not a good performance indicator for possible comparison. Only CPU measures seem a good approach for that purpose.



(a) Considering (P1) and (P2)

(b) Considering (P1), (P2) and (P3) principles

Figure 1.31 – Illustration of LATIN non-linear iterative solver. Waving arrows symbolize approximated search directions in (b).

Local stage (flexible body vs. rigid obstacle)

Given a solution $\mathbf{s} = (\mathbf{u}, \boldsymbol{\lambda})$ verifying the internal balance, kinematic admissibility and static admissibility, the local stage is an updating stage occurring at the contacting interface. At each node, non-linear frictional contact conditions and search direction equations have to be verified. At contacting nodes, search direction equations for both normal and tangential components read:

$$(\mathbf{E}^+) \ : \begin{cases} \hat{\lambda}_N - \lambda_N = k_N(\hat{v}_N - v_N) \\ \hat{\boldsymbol{\lambda}}_T - \boldsymbol{\lambda}_T = k_T(\hat{\mathbf{v}}_T - \mathbf{v}_T) \end{cases} \tag{1.42}$$

$\mathbf{v}$ is the trace of the displacement field $\mathbf{u}$ at the contacting interface such that $\mathbf{v} = \mathbf{B}\mathbf{u}$, k_N and k_T are parameters of the method (details are provided in a next section).

Search direction equations can be equivalently rewritten introducing the gap parameter (for the normal component) and the displacement of the previous time step (for the tangential component, with a superscript $k-1$). This yields to the following equations:

$$\begin{cases} c_N = \lambda_N - k_N(v_N - j) = \hat{\lambda}_N - k_N(\hat{v}_N - j) \\ \mathbf{c}_T = \boldsymbol{\lambda}_T - k_T(\mathbf{v}_T^k - \hat{\mathbf{v}}_T^{k-1}) = \hat{\boldsymbol{\lambda}}_T - k_T(\hat{\mathbf{v}}_T^k - \hat{\mathbf{v}}_T^{k-1}) \end{cases} \tag{1.43}$$

$\mathbf{v}_T^k$ refers to the nodal displacement on the contacting interface in the tangential plane at time step t_k. Then, contact conditions can be easily applied on each term of c_N and $\mathbf{c}_T$. Explicit solutions of the local stage are given in table 1.1 and have to be computed for each node belonging to the contacting interface and for each time step. Hence, the solution of the local stage $\hat{\mathbf{s}} = (\hat{\mathbf{v}}, \hat{\boldsymbol{\lambda}})$ verifies exactly frictional contact conditions and provides a prediction for a given solution $\mathbf{s} = (\mathbf{v}, \boldsymbol{\lambda})$ verifying internal balance in a second step. Moreover, $\hat{\mathbf{s}}$ is computed explicitly.

Normal components	
$\hat{\lambda}_N = \langle c_N \rangle_- \qquad \hat{v}_N = j - \frac{1}{k_N} \langle c_N \rangle_+$	
Tangential components	
Sliding: $\|\mathbf{c}_T\|_2 > \mu \lvert\hat{\lambda}_N\rvert$	Sticking: $\|\mathbf{c}_T\|_2 \leqslant \mu \lvert\hat{\lambda}_N\rvert$
$\mathbf{t} = \mathbf{c}_T / \|\mathbf{c}_T\|_2$ $\hat{\boldsymbol{\lambda}}_T = \mu \lvert\hat{\lambda}_N\rvert \mathbf{t}$ $\hat{\mathbf{v}}_T^k = \hat{\mathbf{v}}_T^{k-1} - \frac{1}{k_T}\Big(\|\mathbf{c}_T\|_2 - \mu \lvert\hat{\lambda}_N\rvert\Big)\mathbf{t}$	$\hat{\boldsymbol{\lambda}}_T = \mathbf{c}_T$ $\hat{\mathbf{v}}_T^k = \hat{\mathbf{v}}_T^{k-1}$

Table 1.1 – Solutions of the local stage (between a flexible solid and a rigid obstacle).

Local stage (flexible body 1 vs. flexible body 2)

Given solutions $\mathbf{s}^{(1)} = (\mathbf{v}^{(1)}, \boldsymbol{\lambda}^{(1)})$ and $\mathbf{s}^{(2)} = (\mathbf{v}^{(2)}, \boldsymbol{\lambda}^{(2)})$ verifying the internal balance, kinematic admissibility and static admissibility for respectively body 1 and 2, search direction equations can be written as above for respectively body 1 and 2. Considering the local coordinate system at the contact interface oriented with the outward normal of the body 1, we define the following known quantities:

$$\begin{cases} \mathbf{c}_N^{(1)} = \boldsymbol{\lambda}_N^{(1)} - k_N \mathbf{v}_N^{(1)} = \hat{\boldsymbol{\lambda}}_N^{(1)} - k_N \hat{\mathbf{v}}_N^{(1)} \\ \mathbf{c}_T^{(1)} = \boldsymbol{\lambda}_T^{(1)} - k_T (\mathbf{v}_T^{k(1)} - \hat{\mathbf{v}}_T^{k-1(1)}) = \hat{\boldsymbol{\lambda}}_T^{(1)} - k_T (\hat{\mathbf{v}}_T^{k(1)} - \hat{\mathbf{v}}_T^{k-1(1)}) \\ \mathbf{c}_N^{(2)} = \boldsymbol{\lambda}_N^{(2)} - k_N \mathbf{v}_N^{(2)} = \hat{\boldsymbol{\lambda}}_N^{(2)} - k_N \hat{\mathbf{v}}_N^{(2)} \\ \mathbf{c}_T^{(2)} = \boldsymbol{\lambda}_T^{(2)} - k_T (\mathbf{v}_T^{k(2)} - \hat{\mathbf{v}}_T^{k-1(2)}) = \hat{\boldsymbol{\lambda}}_T^{(2)} - k_T (\hat{\mathbf{v}}_T^{k(2)} - \hat{\mathbf{v}}_T^{k-1(2)}) \end{cases} \tag{1.44}$$

Previous equations can be combined and rewritten as follows:

$$\begin{cases} C_N = (\mathbf{c}_N^{(1)} - \mathbf{c}_N^{(2)}) \cdot \mathbf{n} + k_N j = -2\hat{\lambda}_N + k_N (\hat{v}_N^{(2)} - \hat{v}_N^{(1)} + j) \\ \mathbf{C}_T = \mathbf{c}_T^{(2)} - \mathbf{c}_T^{(1)} = \hat{\boldsymbol{\lambda}}_T^{(2)} - \hat{\boldsymbol{\lambda}}_T^{(1)} + k_T \Big[\hat{\mathbf{v}}_T^{k(1)} - \hat{\mathbf{v}}_T^{k-1(1)} - (\hat{\mathbf{v}}_T^{k(2)} - \hat{\mathbf{v}}_T^{k-1(2)}) \Big] \end{cases} \tag{1.45}$$

Explicit solutions are given in table 1.2 and have to be computed for each node belonging to the contacting interface of each body and for each time step.

Global stage without separated representation ("full fields formulation")

Given a solution $\hat{\mathbf{s}} = (\hat{\mathbf{v}}, \hat{\boldsymbol{\lambda}})$ from the previous local stage, the global stage consists in finding a space-time solution defined on the whole space domain and time interval verifying the linear constitutive law, kinematic admissibility, static admissibility. We introduce an

<table>
<tr><td colspan="2">Normal components</td></tr>
<tr><td colspan="2">$\hat{\lambda}_N = -\frac{1}{2}\langle \mathcal{C}_N \rangle_-$ $\quad \begin{cases} \hat{\boldsymbol{\lambda}}_N^{(1)} = -\hat{\lambda}_N \mathbf{n} \\ \hat{\boldsymbol{\lambda}}_N^{(2)} = \hat{\lambda}_N \mathbf{n} \end{cases}$ $\quad \begin{cases} \hat{\mathbf{v}}_N^{(1)} = -(\mathbf{c}_N^{(1)} - \hat{\boldsymbol{\lambda}}_N^{(1)})/k_N \\ \hat{\mathbf{v}}_N^{(2)} = -(\mathbf{c}_N^{(2)} - \hat{\boldsymbol{\lambda}}_N^{(2)})/k_N \end{cases}$</td></tr>
<tr><td colspan="2">Tangential components</td></tr>
<tr><td>Sliding: $\|\mathbf{C}_T\|_2 > 2\mu|\hat{\lambda}_N|$</td><td>Sticking: $\|\mathbf{C}_T\|_2 \leqslant 2\mu|\hat{\lambda}_N|$</td></tr>
<tr><td>$\begin{cases} \hat{\lambda}_T = \frac{1}{2}\|\mathbf{C}_T\|_2 \\ \mathbf{t} = -\mathbf{C}_T/\|\mathbf{C}_T\|_2 \end{cases}$ $\quad \begin{cases} \hat{\boldsymbol{\lambda}}_T^{(1)} = \hat{\lambda}_T \mathbf{t} \\ \hat{\boldsymbol{\lambda}}_T^{(2)} = -\hat{\lambda}_T \mathbf{t} \end{cases}$
$\begin{cases} \hat{\mathbf{v}}_T^{k(1)} = \hat{\mathbf{v}}_T^{k-1(1)} + (\hat{\boldsymbol{\lambda}}_T^{(1)} - \mathbf{c}_T^{(1)})/k_T \\ \hat{\mathbf{v}}_T^{k(2)} = \hat{\mathbf{v}}_T^{k-1(2)} + (\hat{\boldsymbol{\lambda}}_T^{(2)} - \mathbf{c}_T^{(2)})/k_T \end{cases}$</td><td>$\begin{cases} \hat{\boldsymbol{\lambda}}_T^{(1)} = -\frac{1}{2}\mathbf{C}_T \\ \hat{\boldsymbol{\lambda}}_T^{(2)} = \frac{1}{2}\mathbf{C}_T \end{cases}$
$\begin{cases} \hat{\mathbf{v}}_T^{k(1)} = \hat{\mathbf{v}}_T^{k-1(1)} + (\hat{\boldsymbol{\lambda}}_T^{(1)} - \mathbf{c}_T^{(1)})/k_T \\ \hat{\mathbf{v}}_T^{k(2)} = \hat{\mathbf{v}}_T^{k-1(2)} + (\hat{\boldsymbol{\lambda}}_T^{(2)} - \mathbf{c}_T^{(2)})/k_T \end{cases}$</td></tr>
</table>

Table 1.2 – Solutions of the local stage (between two flexible bodies).

additional search direction equation for the local stage frictional contact prediction. This stage consists in finding a displacement field and a contact force field $\mathbf{s} = (\mathbf{u}, \boldsymbol{\lambda})$ defined over $\partial_3\Omega \times [0,\ T]$. These fields have to verify the admissibility equations:

$$\begin{cases} \mathbf{K}\mathbf{u} = \mathbf{f}_{\text{ext}} + \mathbf{B}^T\boldsymbol{\lambda} \\ \mathbf{v} = \mathbf{B}\mathbf{u} \end{cases} \tag{1.46}$$

$\mathbf{v}$ is the trace of the displacement field over $\partial_3\Omega \times [0,\ T]$ whereas $\mathbf{u}$ is defined over the whole space-time domain. According to the second principle of the LATIN method, the search direction has to be verified as well. Let us write search direction equations for the whole contacting interface (with $\mathbf{I_d}$ the identity matrix):

$$(\mathbf{E}^-) \ : \ \begin{Bmatrix} \boldsymbol{\lambda}_N \\ \boldsymbol{\lambda}_T \end{Bmatrix} - \begin{Bmatrix} \hat{\boldsymbol{\lambda}}_N \\ \hat{\boldsymbol{\lambda}}_T \end{Bmatrix} = \underbrace{\begin{bmatrix} k_N\mathbf{I_d} & \mathbf{0} \\ \mathbf{0} & k_T\mathbf{I_d} \end{bmatrix}}_{\mathbf{k}} \left(\begin{Bmatrix} \hat{\mathbf{v}}_N \\ \hat{\mathbf{v}}_T \end{Bmatrix} - \begin{Bmatrix} \mathbf{v}_N \\ \mathbf{v}_T \end{Bmatrix} \right) \tag{1.47}$$

All in all, taking into account only the first and the second principle of the LATIN method, leads to solve the following linear system for each time step:

$$\begin{cases} [\mathbf{K} + \mathbf{B}^T\mathbf{k}\mathbf{B}]\mathbf{u} = \mathbf{f}_{\text{ext}} + \mathbf{B}^T(\hat{\boldsymbol{\lambda}} + \mathbf{k}\hat{\mathbf{v}}) \\ \mathbf{v} = \mathbf{B}\mathbf{u} \\ \boldsymbol{\lambda} = \hat{\boldsymbol{\lambda}} + \mathbf{k}(\hat{\mathbf{v}} - \mathbf{v}) \end{cases} \tag{1.48}$$

Note that the operator $[\mathbf{K}+\mathbf{B}^T\mathbf{k}\mathbf{B}]$ is symmetric definite positive and remains constant

along the iterations. For multibodies problems, the global stage occurs for each contacting body considering respective local stage results. Generally, the global stage is the most computationally expensive stage.

Global stage without separated representation ("correction formulation")

The previous formulation can be equivalently rewritten into a correction scheme in space and time. Indeed, given the solution of the previous global stage $\mathbf{s}_{i-1} = (\mathbf{u}_{i-1}, \boldsymbol{\lambda}_{i-1})$, a correction $\boldsymbol{\Delta s} = (\boldsymbol{\Delta u}, \boldsymbol{\Delta \lambda})$ is sought such that the updated solution is $\mathbf{s} = \mathbf{s}_{i-1} + \boldsymbol{\Delta s} = (\mathbf{u}_{i-1} + \boldsymbol{\Delta u}, \boldsymbol{\lambda}_{i-1} + \boldsymbol{\Delta \lambda})$, hence called the correction formulation. As above, updated solution has to fulfill the linear constitutive law, kinematic admissibility, static admissibility. Subtracting appropriate quantities to equations (1.46) yields to the following linear system in order to solve the corrective increments:

$$\begin{cases} \boldsymbol{\Delta u} = \mathbf{u} - \mathbf{u}_{i-1} \\ \boldsymbol{\Delta \lambda} = \boldsymbol{\lambda} - \boldsymbol{\lambda}_{i-1} \\ \mathbf{K}\boldsymbol{\Delta u} = \mathbf{B}^T \boldsymbol{\Delta \lambda} \\ \boldsymbol{\Delta v} = \mathbf{B}\boldsymbol{\Delta u} \\ \boldsymbol{\Delta \lambda} + \mathbf{k}\boldsymbol{\Delta v} - \mathbf{res}_{\text{sd}} = 0 \end{cases} \Leftrightarrow \begin{cases} [\mathbf{K} + \mathbf{B}^T\mathbf{k}\mathbf{B}]\boldsymbol{\Delta u} = \mathbf{B}^T \mathbf{res}_{\text{sd}} \\ \boldsymbol{\Delta v} = \mathbf{B}\boldsymbol{\Delta u} \\ \boldsymbol{\Delta \lambda} = \mathbf{res}_{\text{sd}} - \mathbf{k}\boldsymbol{\Delta v} \end{cases} \tag{1.49}$$

where $\mathbf{res}_{\text{sd}} = \hat{\boldsymbol{\lambda}} - \boldsymbol{\lambda}_{i-1} + \mathbf{k}(\hat{\mathbf{v}} - \mathbf{v}_{i-1})$ is known at this stage. This formulation is particularly suited. Starting from the initial guess $\mathbf{s}_0$ as defined in equation (1.36), this correction scheme builds progressively the complementary corrective solution $\mathbf{s}^* = (\mathbf{u}^*, \boldsymbol{\lambda}^*)$. It is the first step toward the space-time separated representation which will be detailed in 3.

Search direction parameters

Search directions parameters are similar to the penalty parameter in an augmented Lagrangian formulation. Different values for the normal and tangential problem can be chosen. They influence only the convergence rate of the method and optimal values are related to mechanical properties of the studied body [Lad99, CCL99, BC03, TGBNT12]. According to [BC03], close-to-optimum values are searched as follows:

$$k = EL_c \tag{1.50}$$

with E the Young's modulus and L_c a contact characteristic dimension. $k_{N,T}$ have the dimension of a stiffness.

Let us write the pressure distribution $p(\mathbf{x}) = \mathbf{Np}$ with $\mathbf{N}$ the shape functions and $\mathbf{p}$ generalized force. The finite element weak formulation yields to the integrated force as follows $\mathbf{F} = \int_\Omega \mathbf{N}^T\mathbf{Np}\,\mathrm{d}\Omega = \mathbf{Mp}$ with $\mathbf{M}$ a mass-like operator. For linear elements, vector force $\mathbf{F}$ and $\mathbf{p}$ are signed identically, so contact conditions can be applied indifferently

on $\mathbf{F}$ or $\mathbf{p}$ (this is not true for quadratic elements). Optimal search direction parameter values given in (1.50) are such that:

$$\mathbf{F} = \mathbf{M}\mathbf{p} = \mathbf{k}\mathbf{v} \quad \Rightarrow \quad \mathbf{p} = \mathbf{M}^{-1}\mathbf{k}\mathbf{v} = \mathbf{k}'\mathbf{v} \tag{1.51}$$

where $\mathbf{v}$ is a generic displacement field at the contacting boundary. In this dissertation and for a sake of simplicity, considered contact forces are generalized forces $\mathbf{p}$. So, search direction values in following are $\mathbf{k}'$. Optimal values for $\mathbf{k}'$ can be estimated using the following approximation:

$$\begin{cases} k' \sim \frac{E}{L_c'} = \frac{E}{(eL)/(2L_c)} & \text{2D problem} \\ k' \sim \frac{E}{L_c'} = \frac{E}{S/(3L_c)} & \text{3D problem} \end{cases} \tag{1.52}$$

with respectively L and S characteristic length and surface of considered elements and e a characteristic thickness in agreement with the literature [PBG10, RBDG07, TGBNT12].

Initialization and stopping criterion

The iterative LATIN process is initialized with $\mathbf{s}_0$ the following linear contactless elastic solution (first guess):

$$\mathbf{s}_0 = (\mathbf{u}_0, \boldsymbol{\lambda}_0) \quad \text{such that} \quad \begin{cases} \mathbf{K}\mathbf{u}_0 = \mathbf{f}_{\text{ext}} \\ \boldsymbol{\lambda}_0 = \mathbf{0} \end{cases} \tag{1.53}$$

Iterations are stopped once $\mathcal{I} < \varepsilon$ is respected with the convergence indicator (1.54) introduced in [RBDG07]:

$$\mathcal{I} = \max\left(\sqrt{\frac{\|\mathbf{s}_N - \hat{\mathbf{s}}_N\|_\infty^2}{\|\mathbf{s}_N\|_\infty^2 + \|\hat{\mathbf{s}}_N\|_\infty^2}}, \sqrt{\frac{\|\mathbf{s}_T - \hat{\mathbf{s}}_T\|_\infty^2}{\|\mathbf{s}_T\|_\infty^2 + \|\hat{\mathbf{s}}_T\|_\infty^2}} \right)$$
$$\text{with} \quad \|\mathbf{s}_{N,T}\|_\infty^2 = \max_{\partial_3\Omega, t} \left[\frac{1}{k_{N,T}} \lambda_{N,T}^2 + k_{N,T} u_{N,T}^2 \right] \tag{1.54}$$

where $k_{N,T}$ is the search direction parameter. Note that subscripts N and T refer respectively to normal and tangential components. Using maximum norms, the convergence criterion (1.54) is a very tough criterion. A comparison with an other far less restrictive indicators is available in [GDGR14]. In practice, a required precision of 10^{-2} to 10^{-4} for this indicator provides an accurate solution.

Algorithms

Pseudo-codes for both LATIN formulations for frictional contact problems are given in algorithms 1 and 2.

Algorithm 1: LATIN method for frictional contact (full approximated fields formulation).

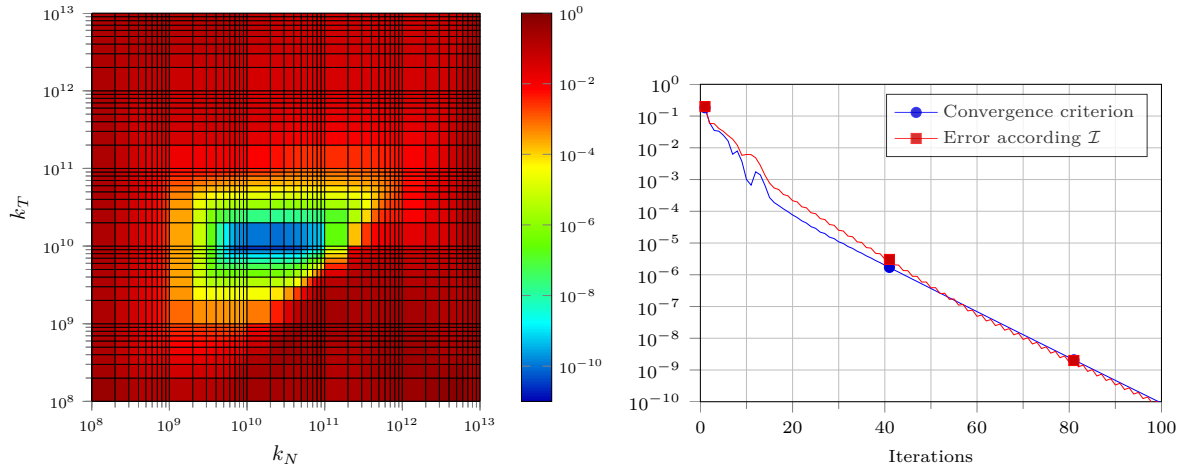
Input:
First guess of the solution $\mathbf{s}_0 = (\mathbf{u}_0, \boldsymbol{\lambda}_0)$
Precision ε
Maximum number of iterations $i_{\max}$
Output: Solution $\mathbf{s} = (\mathbf{u}, \boldsymbol{\lambda})$

Initialization: $i = 0$
while $\mathcal{I} > \varepsilon$ **or** $i < i_{\max}$ **do**
  for $t_{1 \leqslant k \leqslant m}$ **do**
    Solve local stage with $\big(\mathbf{v}(t_k), \boldsymbol{\lambda}(t_k)\big)$ as input to get $\big(\hat{\mathbf{v}}(t_k), \hat{\boldsymbol{\lambda}}(t_k)\big)$ satisfying $\mathcal{R}\big(\hat{\mathbf{v}}(t_k), \hat{\boldsymbol{\lambda}}(t_k)\big) = 0$
  end
  for $t_{1 \leqslant k \leqslant m}$ **do**
    Solve global stage
$$\begin{cases} [\mathbf{K} + \mathbf{B}^T \mathbf{k} \mathbf{B}]\mathbf{u}(t_k) = \mathbf{f}_{\text{ext}}(t_k) + \mathbf{B}^T\big(\hat{\boldsymbol{\lambda}}(t_k) + \mathbf{k}\hat{\mathbf{v}}(t_k)\big) \\ \mathbf{v}(t_k) = \mathbf{B}\mathbf{u}(t_k) \\ \boldsymbol{\lambda}(t_k) = \hat{\boldsymbol{\lambda}}(t_k) + \mathbf{k}\big(\hat{\mathbf{v}}(t_k) - \mathbf{v}(t_k)\big) \end{cases}$$
  end
  Compute convergence criterion $\mathcal{I}$
  $i \leftarrow i + 1$
end
if $i = i_{\max}$ **then** no convergence

Algorithm 2: LATIN method for frictional contact (correction formulation).

Input:
First guess of the solution $\mathbf{s}_0 = (\mathbf{u}_0, \boldsymbol{\lambda}_0)$
Precision ε
Maximum number of iterations $i_{\max}$
Output: Solution $\mathbf{s} = (\mathbf{u}, \boldsymbol{\lambda})$

Initialization: $k = 0$
while $\mathcal{I} > \varepsilon$ **or** $i < i_{\max}$ **do**
  for $t_{1 \leqslant k \leqslant m}$ **do**
    Solve local stage with $(\mathbf{v}(t_k), \boldsymbol{\lambda}(t_k))$ as input to get $(\hat{\mathbf{v}}(t_k), \hat{\boldsymbol{\lambda}}(t_k))$ satisfying $\mathcal{R}(\hat{\mathbf{v}}(t_k), \hat{\boldsymbol{\lambda}}(t_k)) = 0$
  end
  for $t_{1 \leqslant k \leqslant m}$ **do**
    Compute $\mathbf{res}_{\mathrm{sd}}(t_k) = \hat{\boldsymbol{\lambda}}(t_k) - \boldsymbol{\lambda}(t_k) + \mathbf{k}(\hat{\mathbf{v}}(t_k) - \mathbf{v}(t_k))$
    Solve the linear system
$$\begin{cases} [\mathbf{K} + \mathbf{B}^T\mathbf{k}\mathbf{B}]\boldsymbol{\Delta}\mathbf{u}(t_k) = \mathbf{B}^T\mathbf{res}_{\mathrm{sd}}(t_k) \\ \boldsymbol{\Delta}\mathbf{v}(t_k) = \mathbf{B}\boldsymbol{\Delta}\mathbf{u}(t_k) \\ \boldsymbol{\Delta}\boldsymbol{\lambda}(t_k) = \mathbf{res}_{\mathrm{sd}}(t_k) - \mathbf{k}\boldsymbol{\Delta}\mathbf{v}(t_k) \end{cases}$$
    Update solution
$$\begin{cases} \mathbf{u}(t_k) \leftarrow \mathbf{u}(t_k) + \boldsymbol{\Delta}\mathbf{u}(t_k) \\ \mathbf{v}(t_k) \leftarrow \mathbf{B}\mathbf{u}(t_k) \\ \boldsymbol{\lambda}(t_k) \leftarrow \boldsymbol{\lambda}(t_k) + \boldsymbol{\Delta}\boldsymbol{\lambda}(t_k) \end{cases}$$
  end
  Compute convergence criterion $\mathcal{I}$
  $i \leftarrow i + 1$
end
if $i = i_{\max}$ **then** no convergence



(a) Error reached according search direction parameters

(b) LATIN convergence

Figure 1.32 – Convergence and search direction parameters. $k_N = 2 \times 10^{10}$ $\frac{\text{N}}{\text{m}}$ and $k_T = 1 \times 10^{10}$ $\frac{\text{N}}{\text{m}}$ reach an error of $6.5 \cdot 10^{-11}$.

Numerical application

The extrusion problem is solved with the presented LATIN method. First, search direction parameters are set to $k_N = 2 \times 10^{10}$ $\frac{\text{N}}{\text{m}}$ and $k_T = 1 \times 10^{10}$ $\frac{\text{N}}{\text{m}}$. Approximately, these can be deduced from (1.52) considering for L_c the width of the billet: $k = \frac{2EL_c}{eL} = \frac{2 \times 50.8 \times 10^{-3} \times E}{1 \times 0.0127} \sim 5.5 \times 10^{11}$ $\frac{\text{N}}{\text{m}}$. On figure 1.32 is shown error reached after 100 iterations according to different value of search direction parameters. Value provided by formula (1.52) and experimental approach are relatively close. Secondly, the problem is processed with the LATIN method. Both evolution of convergence indicator and error compared to a reference solution (computed with a high level of accuracy) are plotted on figure 1.32.

Convergence diagram on figure 1.32 provides a global "measure" of the convergence but does not pay attention of the different scales of the solution illustrated in section 1.5. In order to know which scales are captured or processed, other tools are required and suggested in the next section.

1.6.3 Convergence analysis through empiric modes

Modal assurance criterion (MAC) diagrams

Given two sets of vectors of same dimension $(\mathbf{X}_i)_1^p$ and $(\mathbf{Y}_i)_1^q$, the MAC matrix [All03] denoted by $\mathbf{M}$ is defined as follows:

$$(\mathbf{M})_{\substack{1 \leqslant i \leqslant p \\ 1 \leqslant j \leqslant q}} = \frac{|\mathbf{X}_i^T \mathbf{Y}_j|^2}{\|\mathbf{X}_i\|^2 \|\mathbf{Y}_j\|^2} \in [0, 1] \tag{1.55}$$

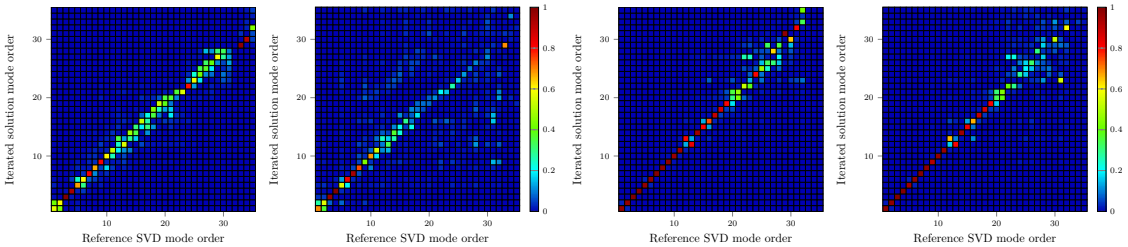
The coefficient M_{ij} measures the correlation between modes $\mathbf{X}_i$ and modes $\mathbf{Y}_j$. If $M_{ij} = 1$, then $\mathbf{X}_i$ and $\mathbf{Y}_j$ are colinear (highly correlated). On the contrary, $M_{ij} = 0$ means that

$\mathbf{X}_i$ and $\mathbf{Y}_j$ are orthogonal (uncorrelated).

LATIN convergence analysis

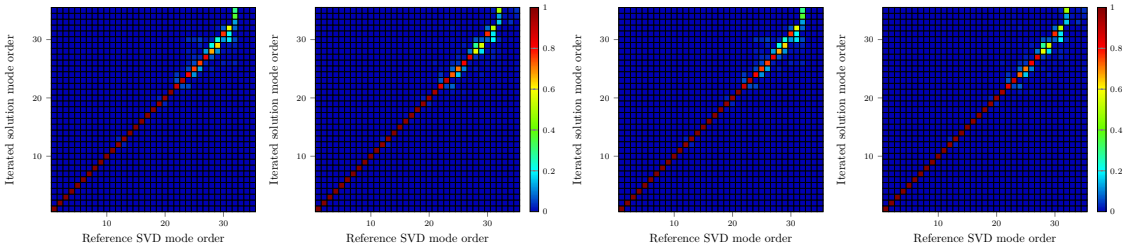
The LATIN method iterates solutions such that it becomes more and more accurate. Intuitively, we suppose that the global scale of the structure is firstly captured active contact zones are coarsely identified. Then, the local scale (*i.e.* contact scale) is solved more precisely. To illustrate this speculation, we propose to use Modal Assurance Criterion (MAC) diagrams. For that purpose, at given numbers of LATIN iteration, iterated solution is analyzed with the SVD providing a set of space-time modes. Given a reference set of modes (*e.g.* ones obtained from the solution), correlation between the obtained set and the reference one is estimated via MAC diagrams. Hence, one is able to identify which scale of the solution is processed or solved. On figure 1.33 are plotted different MAC matrices computed at given numbers of LATIN iteration for the extrusion problem.

According to figure 1.33, the assumption is confirmed. But, one has to keep in mind that the convergence of the LATIN method is also governed by search direction parameters as far as the convergence rate is concerned. Nevertheless, MAC diagrams show that the LATIN method operates separately the different scales of the solution. First, the structural scale (first reference SVD modes) is solved then the local contact scale (higher order ones). So, one can state that the computational work aims at identifying and solving the different scales of the solution successively. Mac diagrams will be used in the next chapter as an efficient way to study the performances of proposed non-linear solvers involving reduction methods.



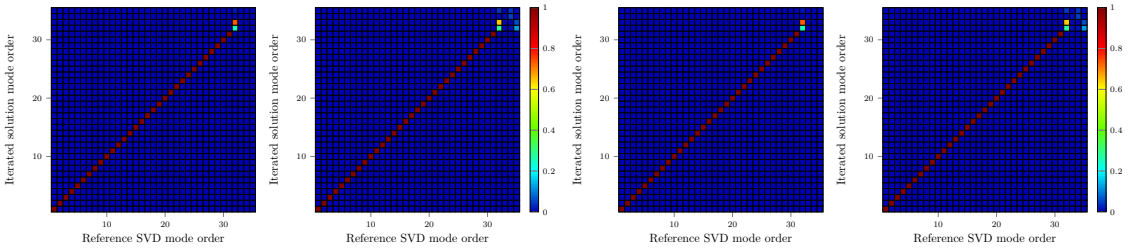
(a) Iteration 1: space (left) and time (right)

(b) Iteration 5: space (left) and time (right)



(c) Iteration 10: space (left) and time (right)

(d) Iteration 15: space (left) and time (right)



(e) Iteration 20: space (left) and time (right)

(f) Iteration 30: space (left) and time (right)

Figure 1.33 – MAC diagrams for the extrusion problem. Color of each mark square corresponds to the value of the correlation between a reference vector and the tested one.

Chapter 2

The FAS/LATIN solver

Abstract

The previous chapter shows a strong scale separability as far as a frictional solution is concerned. This observation fosters greatly multilevel computational techniques such as multigrid methods. This second chapter suggests the FAS/LATIN method which is an accelerated version of the LATIN method relying reduced basis computations. Its basic concept is directly inspired from multigrid methods. So and firstly, main multigrid methods will be briefly reviewed. Then, the FAS/LATIN method introduced. Next, a computational technique for reduced basis is proposed using a surrogate model. Finally, the FAS/LATIN is applied on the previous extrusion problem and resulting performances discussed.

Contents

2.1 Multigrid methods

In the 70s, multilevel computations [Bra77] presaged nowadays well-known multigrid (MG) methods. They belong to an efficient family of both linear and non-linear solvers well suited for problems exhibiting multiscale aspects. The basic idea consists in taking advantage of the smoothing effect observed for some iterative methods (also called relaxation methods) and introducing a hierarchy of discretizations (also called grids). It has been shown in that relaxation methods have a convergence rate depending on the processed wavelength component of the solution (relatively to the discretization): Jacobi's method in [VL00] and conjugate gradient method in [Dur97]. This observation suggests that each wavelength component of the solution has to be tackled differently. Hence, multigrid methods consist in accelerating convergence rate of a relaxation method by computing the different wavelength components of the solution on different discretization levels. Due to their efficiency, multigrid methods are widely used for various applications to solve both linear and non-linear problems. Hereinafter, smoothing effect and the two most widespread MG methods are briefly recalled. But interested readers can refer to a wide literature [Bra05, VL00, Ran08, BGLC10, BHM00]. More recent developments rely on a kind of automation for designing such approaches, mainly on the definition of coarse levels [BMH85, VMB96, Stü01].

2.1.1 Smoothing effect

As a general behavior, relaxation methods have a great ability to diminish short wavelength components of the error contrary to long wavelength ones (relatively to the discretization). This is why they are also called smoothers (*e.g.* Jacobi, Gauss-Seidel, successive over relaxation, conjugate gradients, etc). To exemplify the difficulty to eliminate low frequency component of the error with these solvers, the following two dimensional Laplace equation is solved with a linear Gauss-Seidel method:

$$\begin{cases} \Delta u(x,y) = 0 & \text{on } \Omega = [0,1]^2 \\ u(x,y) = u_p(x,y) & \text{on the boundary } \partial_1\Omega \end{cases} \tag{2.1}$$

with the Dirichlet Boundary condition:

$$u_p(x,y) = \begin{cases} \sinh(\pi)\sin(\pi y) & \text{if } x = 1 \\ 0 & \text{otherwise} \end{cases} \tag{2.2}$$

The continuous exact solution is $v(x,y) = \sinh(\pi x)\sin(\pi y)$ and can be obtained with the variable separation method. For this example, the above problem is solved numerically with the finite difference method on a uniform square Cartesian grid 32×32 leading to a linear system to solve. This system is solved using a direct method providing a numerical solution considered as the reference. The 2-norm of the error between the reference and

the continuous solution is 4.9×10^{-4}. Then, a Gauss-Seidel method is used and iterated solutions with the error map relatively to the reference are plotted on figure 2.1.

The initial guess of the solution is a zero field except at the boundary where Dirichlet conditions are prescribed. It is designed such that the long wavelength component has to be propagated on the domain Ω. Error maps depict clearly that the computational work is mainly concentrated to eliminate a long wavelength component error.

2.1.2 The linear multigrid correction scheme (MG-CS)

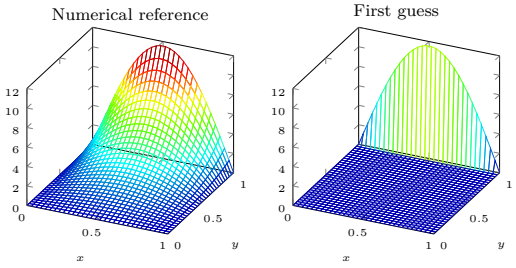
The multigrid correction scheme (MG-CS) method is a linear solver consisting in bringing corrections computed on coarser discretizations after a first group of relaxations denoted by ν_1. Then, coarse discretizations are expected to produce corrections reducing efficiently long wavelength components of the error. To transfer fields, a restriction operator denoted by $\mathbf{R}$ and prolongation (or interpolation) operator $\mathbf{P}$ are defined. Generally, coarse problems are small and easy to solve (possibly with direct solvers). Let h be the characteristic length of the finest grid $\mathcal{G}_h$. Coarse grids are usually designed by using the following rule of thumb for the mesh size $2^p \times h$. According to [VL00], 2^p factors are the optimal ones. Figure 2.2 gives the workflow of the MG-CS and the algorithm 3 gives the pseudo-code for the MG-CS involving two grids.

Algorithm 3: MG-CS linear solver for $\mathbf{Ku} = \mathbf{f}$ (with two grids).

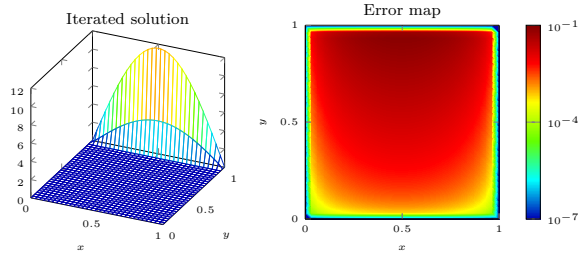
Input: Initial guess $\mathbf{u}_0$
Output: Solution $\mathbf{u}$

Compute residual $\mathbf{res} = \mathbf{f} - \mathbf{K}\mathbf{u}_0$
Initialization $k = 1$
while $\|\mathbf{res}\|/\|\mathbf{f}\| \geqslant$ *precision* **do**
  ν_1 relaxations on $\mathcal{G}_h$ starting from $\mathbf{u}_{k-1}$ leading to solution $\mathbf{u}_{k-2/3}$
  Compute residual on $\mathcal{G}_h$ providing $\mathbf{res} = \mathbf{f} - \mathbf{K}\mathbf{u}_{k-2/3}$
  Restrict residual to $\mathcal{G}_{2h}$ providing $\bar{\mathbf{res}} = \mathbf{R}\mathbf{res}$
  Compute coarse correction on $\mathcal{G}_{2h}$ by solving the coarse problem $\bar{\mathbf{K}}\boldsymbol{\Delta}\bar{\mathbf{u}} = \bar{\mathbf{res}}$
  Coarse correction of the solution on $\mathcal{G}_h$ leading to $\mathbf{u}_{k-1/3} = \mathbf{u}_{k-2/3} + \mathbf{P}\boldsymbol{\Delta}\bar{\mathbf{u}}$
  ν_2 relaxations on $\mathcal{G}_h$ starting from $\mathbf{u}_{k-1/3}$ leading to solution $\mathbf{u}_k$
  Increment cycle $k \leftarrow k + 1$
end

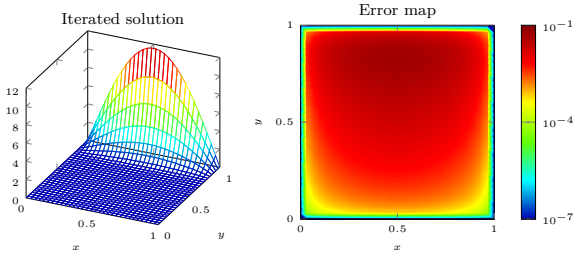
Since coarse correction stage may generate high frequency error, a second relaxation stage (post-smoothing stage) is used. Literature provides multiple possibilities to define restriction and prolongation operator (according to different weighting method). Nonetheless, to preserve the conservation of the internal work they have to verify $\mathbf{R} = \mathbf{P}^T$ (see



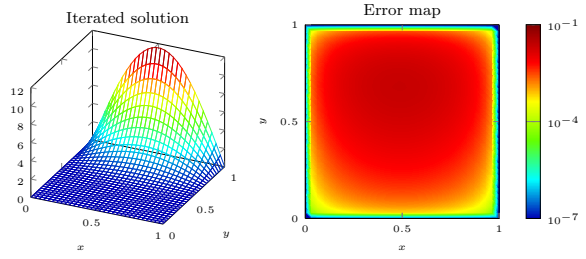
(a) Reference solution and first guess



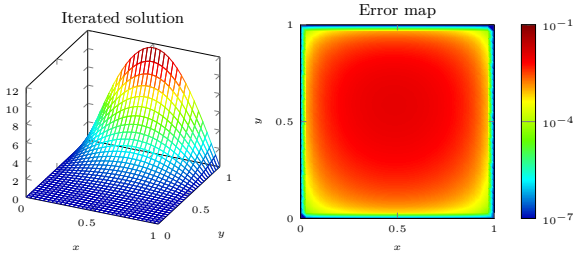
(b) 1 relaxation (2-norm error : 0.85)



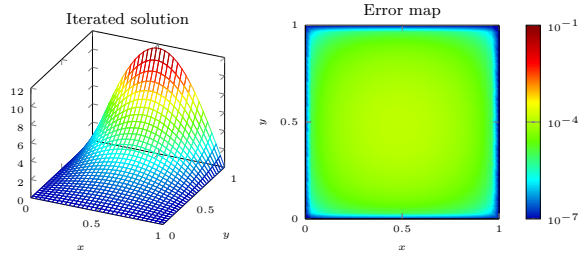
(c) 10 relaxations (2-norm error : 0.66)



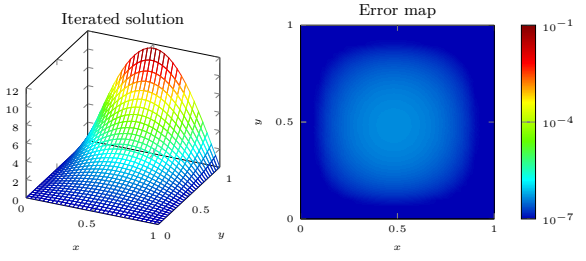
(d) 50 relaxations (2-norm error : 0.36)



(e) 100 relaxations (2-norm error : 0.21)



(f) 500 relaxations (2-norm error : 0.0034)



(g) 1000 relaxations (2-norm error : 2×10^{-5})

Figure 2.1 – The Gauss-Seidel method lacks in diminishing the low frequency component of the error.

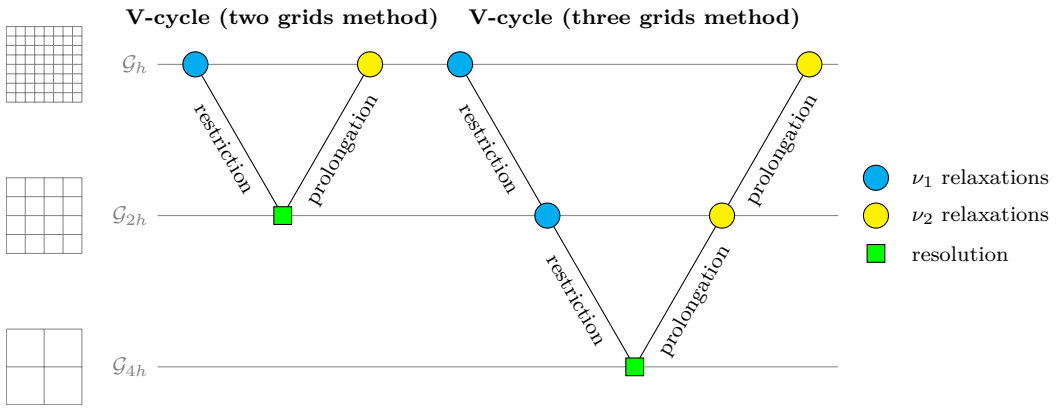


Figure 2.2 – The MG-CS workflow involving two or three discretizations. Such a process is the V-cycle. Other cycling versions like W-cycle were also proposed.

[Ran08]). Coarse operators are generally also defined with the Galerkin projected operator $\bar{\mathbf{K}} = \mathbf{RKP}$.

2.1.3 The non-linear multigrid full approximation scheme (MG-FAS)

Multigrid methods also tackles non-linear problems with the multigrid full approximation scheme (MG-FAS). The overall process remains the same as for the MG-CS (workflow on figure 2.2). But full approximations $\bar{\mathbf{u}}$ are processed on the different grids instead of corrective increments $\Delta\bar{\mathbf{u}}$, hence the eponymous method. Due to a non-linear operator denoted formally by L, the property $L(\mathbf{u}+\mathbf{v}) = L(\mathbf{u}) + L(\mathbf{v})$ no longer stands and the coarse problem has to be reformulated. Moreover, one has to remark that relaxation stages (and usually the coarse problem as well) are non-linear. Quasi-Newton methods (*e.g.* BFGS) are often used as smoothers. Let us denote by L a formal non-linear operator on the finest grid and $\bar{L}$ on the coarser one. The main difficulty lies in the definition of the coarse problem aiming to diminish the error on the finest grid. Let $\mathbf{u}$ be the solution on the finest grid. Then, it verifies:

$$L(\mathbf{u}) = \mathbf{f} \tag{2.3}$$

Starting from a guess $\mathbf{u}_{k-1}$, relaxation methods are able to provide an iterated solution denoted by $\mathbf{u}_{k-2/3}$ yielding to the residual vector:

$$\mathbf{res} = \mathbf{f} - L(\mathbf{u}_{k-2/3}) \tag{2.4}$$

The error vector is defined as:

$$\boldsymbol{\varepsilon} = \mathbf{u} - \mathbf{u}_{k-2/3} \tag{2.5}$$

Substituting equation (2.4) and (2.5) into (2.3) reads:

$$L(\mathbf{u}_{k-2/3} + \boldsymbol{\varepsilon}) = L(\mathbf{u}_{k-2/3}) + \mathbf{res} \tag{2.6}$$

Then, equation (2.6) is expected to be solved on the coarse grid:

$$\bar{L}(\bar{\mathbf{u}}) = \bar{L}(\bar{\mathbf{R}}\mathbf{u}_{k-2/3}) + \mathbf{R}\mathbf{res} \tag{2.7}$$

Once an approximation for $\bar{\mathbf{u}}$ obtained, the solution $\mathbf{u}_{k-2/3}$ is corrected providing an updated iterated solution $\mathbf{u}_{k-1/3}$:

$$\mathbf{u}_{k-1/3} = \mathbf{u}_{k-2/3} + \mathbf{P}(\bar{\mathbf{u}} - \bar{\mathbf{R}}\mathbf{u}_{k-2/3}) \tag{2.8}$$

Hence, the MG-FAS estimates a correction $\mathbf{P}(\bar{\mathbf{u}}-\bar{\mathbf{R}}\mathbf{u}_{k-2/3})$ for the error $\boldsymbol{\varepsilon}$. Restriction and prolongation operators defined for MG-CS are also available for MG-FAS. Nevertheless, one can remark that not only residual vector **res** is restricted (by operator $\mathbf{R}$) but also $\mathbf{u}_{k-2/3}$ (by operator $\bar{\mathbf{R}}$). $\bar{\mathbf{R}}$ and $\mathbf{R}$ can be chosen differently [Ran08]. The corresponding MG-FAS pseudo-code is given on algorithm 4.

Algorithm 4: MG-FAS non-linear solver for $L(\mathbf{u}) = \mathbf{f}$ (with two grids).

Input: Initial guess $\mathbf{u}_0$
Output: Solution $\mathbf{u}$

Compute residual $\mathbf{res} = \mathbf{f} - L(\mathbf{u}_0)$
Initialization $k = 1$
while $\|\mathbf{res}\|/\|\mathbf{f}\| \geqslant$ *precision* **do**
  ν_1 relaxations on $\mathcal{G}_h$ starting from $\mathbf{u}_{k-1}$ leading to solution $\mathbf{u}_{k-2/3}$
  Compute residual on $\mathcal{G}_h$ providing $\mathbf{res} = \mathbf{f} - L(\mathbf{u}_{k-2/3})$
  Compute coarse solution on $\mathcal{G}_{2h}$ by solving $\bar{L}(\bar{\mathbf{u}}) = \mathbf{R}\mathbf{r}\bar{\mathbf{e}}\mathbf{s} + \bar{L}(\bar{\mathbf{R}}\mathbf{u}_{k-2/3})$
  Coarse correcting on $\mathcal{G}_h$ leading to $\mathbf{u}_{k-1/3} = \mathbf{u}_{k-2/3} + \mathbf{P}(\bar{\mathbf{u}} - \bar{\mathbf{R}}\mathbf{u}_{k-2/3})$
  ν_2 relaxations on $\mathcal{G}_h$ starting from $\mathbf{u}_{k-1/3}$ leading to solution $\mathbf{u}_k$
  Increment cycle $k \leftarrow k + 1$
end

For non-linear problems, other applications based on MG methods were also proposed. For example, the full multigrid method (FMG) [BHM00, BL11] consists in computing relevant approximations computed on coarse grids (cheap computations) to start relaxation sweeps on finer ones. Such a method seems to be nowadays one of the most efficient multigrid approach.

2.2 The FAS/LATIN algorithm

The scale separability for frictional contact solutions observed in section 1.5 fosters multiscale and multilevel strategies. These are general methods to solve physical problems which have important features at multiple scales (both spatial and temporal) [Ste08, Hor10]. For instance, in [PBG10], a cracked body is studied taking into account frictional contacts between crack faces. The behavior at interfaces (crack faces) is treated as an

autonomous problem (own discretization, constitutive law, internal variables) and then linked in a weak sense to the global problem. In [KPB11], a local/global strategy is used by solving the global problem in a reduced subspace while the local problem (where a localized damage phenomenon occurs) is fully resolved. Then, both of the global and local problems are coupled. For contact problems in [Leb89, LRLG91, Gre95, LRR07], MG-methods were proposed.

On the one hand, multigrid methods propose a relevant approach by adapting the computational support (*i.e.* grids) to the different scales of the solution. On the other hand, the LATIN convergence analysis shows that the computational work focuses different scales of the solution (see section 1.6.3) during its iterative process. Hence, an analogy with the multigrid basic idea is proposed by adapting the computational support to processed scales. Global scale empiric modes could be considered as generating coarse grid subspace and higher order ones as finer grids. Therefore, by computing coarse corrections on coarse modes, it is expected to accelerate the LATIN method on those scales. Coarse space-time corrections are computed in a reduced subspace requiring a low computational effort. Then, they are transferred to the full dimensional model to accelerate the convergence rate of the iterative LATIN solver (considered as the smoother). The MG-FAS formulation is used to achieve the accelerated FAS/LATIN method. Doing so, the overall algorithm consists in:

1. a few iterations (*i.e.* ν_1 relaxations) with the non-linear LATIN solver;
2. computing a coarse correction in a reduced subspace spanned by given space vectors;
3. a few iterations (*i.e.* ν_2 relaxations) with the non-linear LATIN solver.

To transfer quantities between the reduced subspace to the full dimensional space, analogs of intergrid operators (restriction and prolongation) are defined:

- $\mathbf{P}$ is the prolongation operator (space vectors) for displacement field (primal)
- $\bar{\mathbf{P}}$ is the prolongation operator (space vectors) for contact force field (dual)
- $\bar{\mathbf{R}}$ is the reduction operator for dual quantities
- $\mathbf{R}$ is the reduction operator for primal quantities

As mentioned in [Ran08], reduction and prolongation operators have to fulfill energy conservation condition. Consequently, $\mathbf{R} = \mathbf{P}^T$ has to be ensured.

2.2.1 Full dimensional problem (fine problem)

LATIN iterations are considered as relaxation sweeps on the finest description of the model and converge to the solution of the reference problem. Contrary to the MG-FAS formulation, the non-linear operator L and its arising residual **res** are not explicitly given.

An analog for **res** as an error estimator could be the convergence criterion defined in (1.54) used for the LATIN method. At the end of the first (respectively second) relaxation stage, the solution $\mathbf{s}_{k-2/3} = (\mathbf{u}_{k-2/3}, \boldsymbol{\lambda}_{k-2/3})$ (respectively $\mathbf{s}_k = (\mathbf{u}_k, \boldsymbol{\lambda}_k)$) is produced.

2.2.2 Reduced problem (coarse problem)

A mix between the LATIN solver and the MG-FAS formulation is proposed. The coarse problem is expected to provide a relevant solution computed in a reduced subspace for both displacement field and contact force field issued. Then and as prescribed by the MG-FAS method, one looks for a full approximated coarse solution $\check{\mathbf{s}} = (\check{\mathbf{u}}, \check{\boldsymbol{\lambda}})$. It is computed such that internal balance and frictional contact conditions are verified as well as possible (up to the accuracy of the reduced basis). For that purpose, a two-stage iterative process with a non-linear local stage and a "reduced" global stage is formulated. This reduced LATIN is a non-linear iterative solver formulated in such a way that the solution of the global stage is processed in a subspace spanned with a given reduced basis.

Local stage

The local stage of the coarse problem consists in computing $(\hat{\mathbf{u}}, \hat{\boldsymbol{\lambda}})$ satisfying frictional contact conditions (*i.e.* $\mathcal{R}(\hat{\mathbf{u}}, \hat{\boldsymbol{\lambda}}) = 0$ as denoted in equation (1.20)) given a coarse solution $(\check{\mathbf{u}}, \check{\boldsymbol{\lambda}})$ as input. Frictional contact conditions are chosen to be strictly verified (*i.e.* not written into a reduced basis as in [HSW12]). Thus, the local stage described in the LATIN method (see section 1.6.2) is reused.

Reduced global stage

Given $(\hat{\mathbf{u}}, \hat{\boldsymbol{\lambda}})$ and a previous guess for coarse solution $(\check{\mathbf{u}}, \check{\boldsymbol{\lambda}})$, one looks for increments $\boldsymbol{\Delta}\check{\mathbf{u}}$ and $\boldsymbol{\Delta}\check{\boldsymbol{\lambda}}$ such that the accuracy of the updated coarse solution $(\check{\mathbf{u}} + \boldsymbol{\Delta}\check{\mathbf{u}}, \check{\boldsymbol{\lambda}} + \boldsymbol{\Delta}\check{\boldsymbol{\lambda}})$ is improved (see correction scheme LATIN formulation on section 1.6.2) :

$$\begin{cases} \mathbf{K}\boldsymbol{\Delta}\check{\mathbf{u}} = \mathbf{B}^T\boldsymbol{\Delta}\check{\boldsymbol{\lambda}} \\ \boldsymbol{\Delta}\check{\mathbf{v}} = \mathbf{B}\boldsymbol{\Delta}\check{\mathbf{u}} \\ \boldsymbol{\Delta}\check{\boldsymbol{\lambda}} + \mathbf{k}\boldsymbol{\Delta}\check{\mathbf{v}} - \mathbf{res}_{\text{sd}} = 0 \end{cases} \tag{2.9}$$

where $\mathbf{res}_{\text{sd}} = \hat{\boldsymbol{\lambda}} - \check{\boldsymbol{\lambda}} + \mathbf{k}(\hat{\mathbf{v}} - \check{\mathbf{v}})$ is known at this stage. Moreover, increments are represented using a space-time separated form with p prescribed space vectors (*i.e.* are cast into $\mathbf{P}$):

$$\begin{cases} \boldsymbol{\Delta}\check{\mathbf{u}} = \sum_i^p \mathbf{P}_i\psi_i(t) \\ \boldsymbol{\Delta}\check{\boldsymbol{\lambda}} = \sum_i^p \bar{\mathbf{P}}_i\theta_i(t) \end{cases} \tag{2.10}$$

Unknowns of the coarse global stage are time vectors $\psi(t)$ and $\theta(t)$. Taking into account this representation, the linear system (2.9) becomes overdetermined. Therefore,

corrective increments are chosen such that internal balance is preserved and the search direction equation approximated. Doing so, the following conditions have to be fulfilled:

$$\mathbf{K}\boldsymbol{\Delta}\check{\mathbf{u}} = \mathbf{B}^T\boldsymbol{\Delta}\check{\boldsymbol{\lambda}} \;\Rightarrow\; \forall t \in [0,T] : \begin{cases} \mathbf{K}\mathbf{P} = \mathbf{B}^T\bar{\mathbf{P}} \\ \psi(t) = \theta(t) \end{cases} \tag{2.11}$$

According to the above conditions, fixing $\mathbf{P}$ (basis of space vectors for displacement field) prescribes $\bar{\mathbf{P}}$ (basis of space vectors for contact force field) and unknowns time functions are equal. Let us define a trial function $\mathbf{u}^\star = \mathbf{P}_j\psi_j^\star(t)$ associated to the j^{th} space vector. The corresponding weakened search direction equation yields to:

$$\begin{aligned} \forall \psi_j^\star : &\int_0^T \psi_j^\star(t)\mathbf{P}_j^T\Big[\mathbf{B}^T\Big(\sum_{i=1}^{p}\bar{\mathbf{P}}_i\theta_i(t)\Big) + \mathbf{B}^T\mathbf{k}\mathbf{B}\Big(\sum_{i=1}^{p}\mathbf{P}_i\psi_i(t)\Big) - \mathbf{B}^T\mathbf{res}_{\text{sd}}\Big]\,\mathrm{d}t \\ &= \int_0^T \psi_j^\star(t)\mathbf{P}_j^T\Big[(\mathbf{K}+\mathbf{B}^T\mathbf{k}\mathbf{B})\Big(\sum_{i=1}^{p}\mathbf{P}_i\psi_i(t)\Big) - \mathbf{B}^T\mathbf{res}_{\text{sd}}\Big]\,\mathrm{d}t = 0 \end{aligned} \tag{2.12}$$

If one writes all weak forms associated to each prescribed space vectors, one will obtain a linear system with time functions as unknowns. It can be cast into the following matrix form:

$$[\mathbf{R}(\mathbf{K}+\mathbf{B}^T\mathbf{k}\mathbf{B})\mathbf{P}]\boldsymbol{\psi}^T = \mathbf{R}\mathbf{B}^T\mathbf{res}_{\text{sd}} \tag{2.13}$$

The size of the above linear system is equal to the number of considered space modes. Moreover, choosing a suited orthonormality condition (related to the operator $\mathbf{K}+\mathbf{B}^T\mathbf{k}\mathbf{B}$), it can be 1-diagonalized. Once solved for $\boldsymbol{\psi}$, the coarse solution can be updated and reads:

$$\check{\mathbf{s}} = \begin{cases} \check{\mathbf{u}} \leftarrow \check{\mathbf{u}} + \mathbf{P}\boldsymbol{\psi}^T \\ \check{\boldsymbol{\lambda}} \leftarrow \check{\boldsymbol{\lambda}} + \bar{\mathbf{P}}\boldsymbol{\psi}^T \end{cases} \tag{2.14}$$

Initialization

The coarse problem is initialized with a time function deduced by restricting the solution of the previous relaxation stage:

$$\boldsymbol{\xi}_0 = \bar{\mathbf{R}}(\mathbf{u}_{k-2/3} - \mathbf{u}_0) \tag{2.15}$$

Then, the initial coarse solution is $\check{\mathbf{s}}_0 = (\check{\mathbf{u}} = \mathbf{u}_0 + \mathbf{P}\boldsymbol{\xi}_0^T, \check{\boldsymbol{\lambda}} = \boldsymbol{\lambda}_0 + \bar{\mathbf{P}}\boldsymbol{\xi}_0^T)$.

Terminating the coarse problem iteration and FAS correction stage

Once the coarse problem iterations are terminated, a full approximated coarse solution can always be written in the following form:

$$\check{\mathbf{s}} = \begin{cases} \check{\mathbf{u}} = \mathbf{u}_0 + \mathbf{P}\boldsymbol{\xi}^T \\ \check{\boldsymbol{\lambda}} = \boldsymbol{\lambda}_0 + \bar{\mathbf{P}}\boldsymbol{\xi}^T \end{cases} \tag{2.16}$$

because each iteration on the coarse problem is restricted to a reduced space basis (*i.e.* each coarse iteration corresponds to an update of the time function $\boldsymbol{\xi}$ adding an increment ψ). The solution of the previous relaxation stage is afterwards corrected according to the MG-FAS coarse correcting stage:

$$\mathbf{s}_{k-1/3} = \begin{cases} \mathbf{u}_{k-1/3} = \mathbf{u}_{k-2/3} + \mathbf{P}[\boldsymbol{\xi}^T - \bar{\mathbf{R}}(\mathbf{u}_{k-2/3} - \mathbf{u}_0)] \\ \boldsymbol{\lambda}_{k-1/3} = \boldsymbol{\lambda}_{k-2/3} + \bar{\mathbf{P}}[\boldsymbol{\xi}^T - \bar{\mathbf{R}}(\mathbf{u}_{k-2/3} - \mathbf{u}_0)] \end{cases} \tag{2.17}$$

One has to keep in mind that the coarse problem does generally not converge to the reference solution. It depends on the accuracy of the considered subspace (which may or may not span the reference solution). The provided coarse solution $\check{\mathbf{s}}$ is the one satisfying as well as possible internal balance, frictional conditions and space-time representation with prescribed space vectors. The convergence criterion defined in equation (1.54) measuring the error between the local and the reduced global solutions may stagnates (*i.e.* the prescribed space vectors does not allow further improvement as far as global and local behavior are concerned). Moreover, the coarse problem has to produce a relevant solution to efficiently correct the solution of relaxation stages. Hence, the coarse problem iterations are terminated once the convergence criterion stagnates or once the quality of the coarsely corrected solution $\mathbf{s}_{k-1/3}$ does not improve.

One can also remark that the FAS/LATIN method tackles the part $\mathbf{s}^*$ defined in equation (1.36) of the sought solution $\mathbf{s}$.

Pseudocode of the FAS/LATIN

The pseudocode of the suggested FAS/LATIN is in algorithm 5. FAS/LATIN cycling process is firstly started with the coarse stage in order to provide an improved initial guess $\mathbf{s}_0$ for the first relaxation stage. This trick is known as the full multigrid method and turns out to be efficient for general non-linear problems as mentioned in [LRR07]. In [GDGR14], a FAS/LATIN formulation slightly different than the one previously is given. The main difference lies in the definition of the coarse problem. In [GDGR14], the coarse problem is defined in such a way the internal balance is not ensured (but verified in a weak sense) and search direction equations satisfied arising nonetheless important accelerations. The new suggested and improved FAS/LATIN provides same performances but is more robust. Following observations and discussions lead to same conclusions for both FAS/LATIN formulations.

2.3 Computational methods for space modes

To perform the coarse problem, space vectors defining operators $\mathbf{P}$, $\bar{\mathbf{P}}$, $\mathbf{R}$ have to be defined. They have to span an appropriate subspace wherein relevant solutions are computed. This subspace has to approximate the part $\mathbf{s}^*$ (defined on equation (1.36)) of the

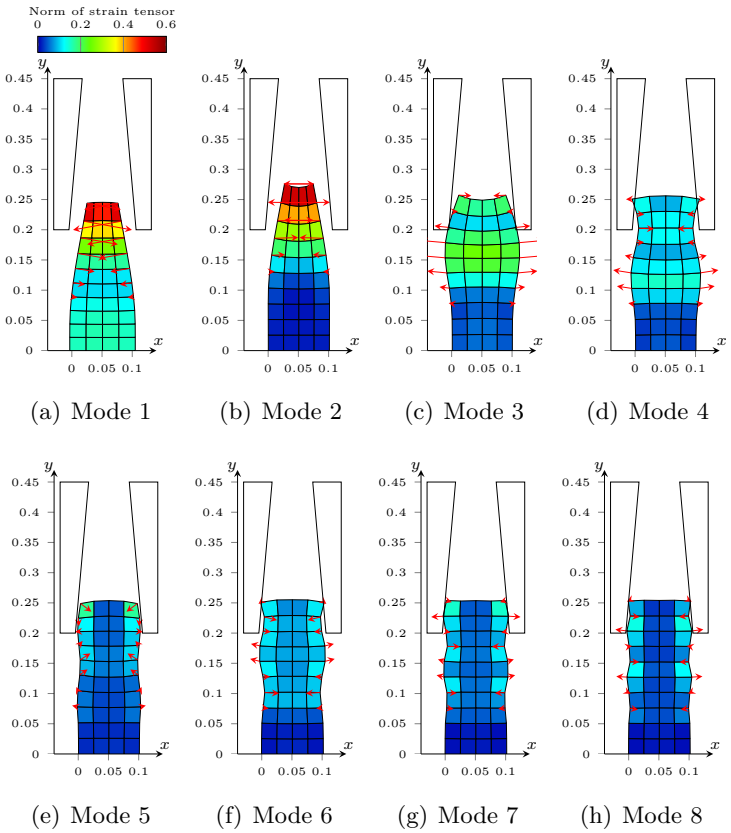
Algorithm 5: The FAS/LATIN method .

Input:
First guess of the solution $\mathbf{s}_0 = (\mathbf{u}_0, \boldsymbol{\lambda}_0)$
Precision ε
Maximum number of FAS/LATIN cycles $k_{\max}$ and iterations of coarse stage $i_{\max}$
Number of relaxations ν_1 and ν_2

Initialization $k = 0$
while $\mathcal{I} > \varepsilon$ **or** $k < k_{\max}$ **do**
  if $k = 0$ **then** $\mathbf{s} = \mathbf{s}_0$ **else** $\mathbf{s} = \mathbf{s}_{k-1}$
  `** 1st relaxation stage`
  if $k > 0$ **then**
    Perform ν_1 LATIN sweeps to get $\mathbf{s}_{k-2/3} = (\mathbf{u}_{k-2/3}, \boldsymbol{\lambda}_{k-2/3})$ and convergence criterion $\mathcal{I}_{k-2/3}$
  end
  `** Coarse stage`
  Create or update operators $\bar{\mathbf{R}}$, $\mathbf{P}$, $\bar{\mathbf{P}} = \mathbf{K}\mathbf{P}$
  Compute initial time vector $\boldsymbol{\xi}^T = \bar{\mathbf{R}}(\mathbf{u}_{k-2/3} - \mathbf{u}_0)$
  Compute initial coarse solution $\check{\mathbf{s}} = (\mathbf{u}_0 + \mathbf{P}\boldsymbol{\xi}^T, \boldsymbol{\lambda}_0 + \bar{\mathbf{P}}\boldsymbol{\xi}^T)$
  Initialization $i = 0$
  while $i < i_{\max}$ **do**
    for $t_{1 \leqslant k \leqslant m}$ **do**
      Solve local stage with $(\mathbf{v}(t_k), \boldsymbol{\lambda}(t_k))$ as input to get $(\hat{\mathbf{v}}(t_k), \hat{\boldsymbol{\lambda}}(t_k))$ satisfying $\mathcal{R}(\hat{\mathbf{v}}(t_k), \hat{\boldsymbol{\lambda}}(t_k)) = 0$
    end
    for $t_{1 \leqslant k \leqslant m}$ **do**
      Compute $\mathbf{res}_{\mathrm{sd}}(t_k) = \hat{\boldsymbol{\lambda}}(t_k) - \boldsymbol{\lambda}(t_k) + \mathbf{k}(\hat{\mathbf{v}}(t_k) - \mathbf{v}(t_k))$
      Compute time functions by solving $[\mathbf{R}(\mathbf{K} + \mathbf{B}^T\mathbf{k}\mathbf{B})\mathbf{P}]\boldsymbol{\psi}^T(t_k) = \mathbf{R}\mathbf{B}^T\mathbf{res}_{\mathrm{sd}}(t_k)$
    end
    Update coarse time functions $\boldsymbol{\xi} \leftarrow \boldsymbol{\xi} + \boldsymbol{\psi}$
    Compute coarse solution $\check{\mathbf{s}} = (\mathbf{u}_0 + \mathbf{P}\boldsymbol{\psi}^T, \boldsymbol{\lambda}_0 + \bar{\mathbf{P}}\boldsymbol{\psi}^T)$
    Compute coarsely corrected solution

$$\mathbf{s}_{k-1/3} = \begin{cases} \mathbf{u}_{k-1/3} = \mathbf{u}_{k-2/3} + \mathbf{P}[\boldsymbol{\xi}^T - \bar{\mathbf{R}}(\mathbf{u}_{k-2/3} - \mathbf{u}_0)] \\ \boldsymbol{\lambda}_{k-1/3} = \boldsymbol{\lambda}_{k-2/3} + \bar{\mathbf{P}}[\boldsymbol{\xi}^T - \bar{\mathbf{R}}(\mathbf{u}_{k-2/3} - \mathbf{u}_0)] \end{cases}$$

    Compute convergence criterion $\mathcal{I}_{k-1/3} = \mathcal{I}(\mathbf{s}_{k-1/3})$
    if $\mathcal{I}_{k-1/3} \geqslant \mathcal{I}_{k-2/3}$ **then** $i = i_{\max}$ **else** $i \leftarrow i + 1$
  end
  `** 2nd relaxation stage`
  Perform ν_2 LATIN sweeps to get $\mathbf{s}_k = (\mathbf{u}_k, \boldsymbol{\lambda}_k)$ and convergence criterion $\mathcal{I}$
  Increment cycles $k \leftarrow k + 1$
end



(a) Mode 1 (b) Mode 2 (c) Mode 3 (d) Mode 4

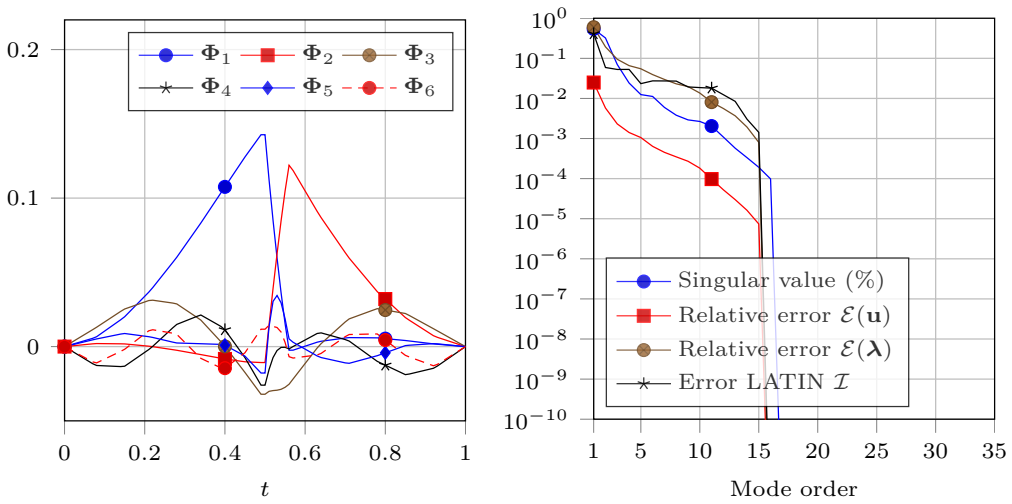
(e) Mode 5 (f) Mode 6 (g) Mode 7 (h) Mode 8

Figure 2.3 – SVD space modes of coarse $\mathbf{u}^*$ scaled by a corresponding factor $0.2\sqrt{\sigma_i}$. The color map refers to the norm of the tensor strain field $\|\boldsymbol{\varepsilon}\| = \sqrt{\boldsymbol{\varepsilon} : \boldsymbol{\varepsilon}}$ and red arrows correspond to deduced nodal contact force.

solution . For instance, a reduced basis including SVD space modes computed from the solution in section 1.5.1 is well-suited. Such a basis will firstly be investigated to exemplify its influence on the LATIN solver (see numerical application on section 2.4). Then, other kinds of vectors will be used.

2.3.1 Surrogate model

Some space modes may be obtained from surrogate computational models (coarse discretizations, analytic solution, etc.). For instance, coarse space-time discretizations suffice to capture the global behavior of the solution (*i.e.* at structural scale). Then, some of these space vectors can be reused for the FAS/LATIN on the reference computational model. This method is illustrated on the extrusion problem depicted on figure 1.7. A two times coarser space-time discretization is considered for the coarse computational model. Once it is solved, SVD modes of the solution field $\mathbf{s}^*$ are computed (see figures 2.3 and 2.4).



(a) Time modes

(b) Amplitudes and errors

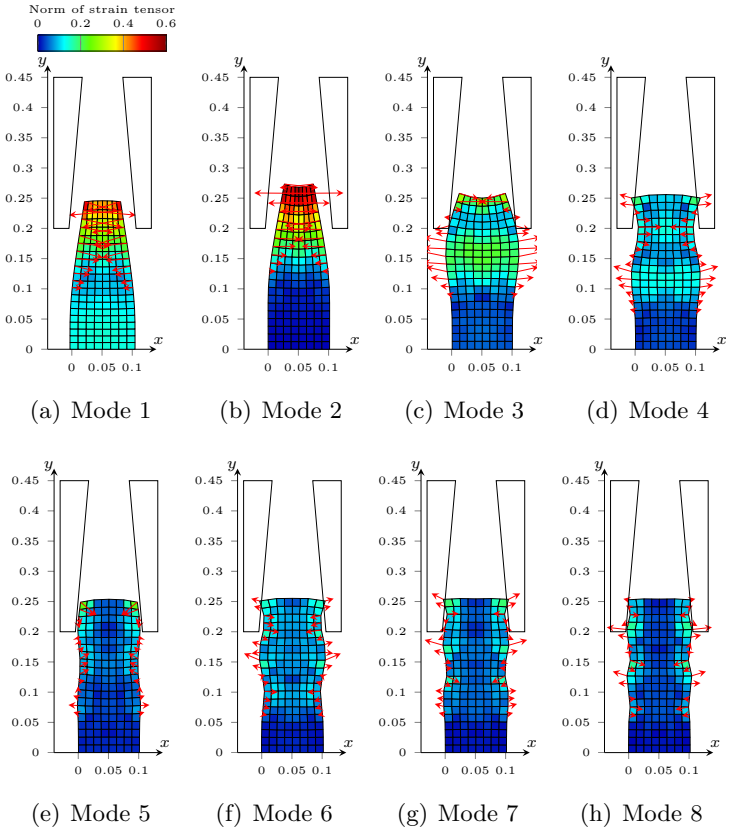
Figure 2.4 – SVD time modes of $\mathbf{u}^*$ scaled by a corresponding factor $0.2\sqrt{\sigma_i}$, singular value amplitudes and rebuilding errors according the number of considered modes.

Remarkably, both space and time modes computed from the coarse computational model show great similarities with ones computed on the reference computational model in section 1.5.1. One can also observe the influence of the discretization on the number of modes. Indeed, the twice coarser solution requires roughly twice less space-time modes to be rebuilt exactly. It can be also seen that after the 7^{th} order, the characteristic scale of spatial modes is strongly related to the mesh of the solid (*i.e.* the discretization scale). The next step consists in transferring space modes on the reference computational model mesh. For that purpose, several mesh transferring methods for displacement fields could be used (*e.g.* collocation method, mortar-based techniques [DB06], etc.). Herein, using a collocation method, space displacement modes on figure 2.3 are interpolated on the reference mesh. Finally, an additional re-orthogonalization step for interpolated space modes is applied and space modes are ready to use for the FAS/LATIN method.

Interpolated space modes (see figure 2.5) are outstandingly close to reference modes computed in section 1.5.1. They capture successfully some scales of the exemplified problems (structural scale, some contact scales). Contact force modes can also be deduced using relationship (1.39). One can remark that they are globally well captured but not as well as displacement modes (interpolation error).

2.3.2 Toward an hybrid a posteriori/a priori approach

The previous method to compute space modes is related to an a posteriori approach. In other words, space modes are computed during an offline phase. Then the online phase occurs with the FAS/LATIN running with a prescribed subspace spanned by given



(a) Mode 1 (b) Mode 2 (c) Mode 3 (d) Mode 4

(e) Mode 5 (f) Mode 6 (g) Mode 7 (h) Mode 8

Figure 2.5 – Interpolated SVD space modes of coarse $\mathbf{u}^*$ scaled by a corresponding factor $0.2\sqrt{\sigma_i}$. The color map refers to the norm of the tensor strain field $\|\boldsymbol{\varepsilon}\| = \sqrt{\boldsymbol{\varepsilon} : \boldsymbol{\varepsilon}}$ and red arrows correspond to deduced nodal contact force.

space modes. Another method consists in computing space modes on-the-fly (*i.e.* during the FAS/LATIN cycling process). Doing so, the FAS/LATIN method is able to start from scratch (*i.e.* from an empty basis), or given a reduced basis, on-the-fly computed vectors may complete it. Such a calculation is closely related to a priori approach and is hereinafter presented.

Let us assume first that a sample of space vectors for $\mathbf{P}$ is already fixed ("a posteriori space vectors"). Computed on-the-fly "a priori space vectors" have to span an orthogonal subspace to a posteriori ones. This subspace can be obtained by computing a reduced basis of the following "orthogonal displacement field":

$$\mathbf{u}^{\perp} = (\mathbf{u} - \mathbf{u}_0) - \mathbf{P}\mathbf{P}^T(\mathbf{u} - \mathbf{u}_0) \tag{2.18}$$

One has to be advised that such a computation can be expensive. Consequently, a compromise has to be reach between the relevancy of the coarse stage and its online computational cost induced by such vectors. Once $\mathbf{P}$ is fixed, $\bar{\mathbf{P}}$ is deduced according. It would be possible to first compute $\bar{\mathbf{P}}$ (whose size is the number of contact points whereas $\mathbf{P}$ size is the number of degrees of freedom) and then deduce $\mathbf{P}$ solving a linear system.

2.4 Application of the FAS/LATIN method and discussion

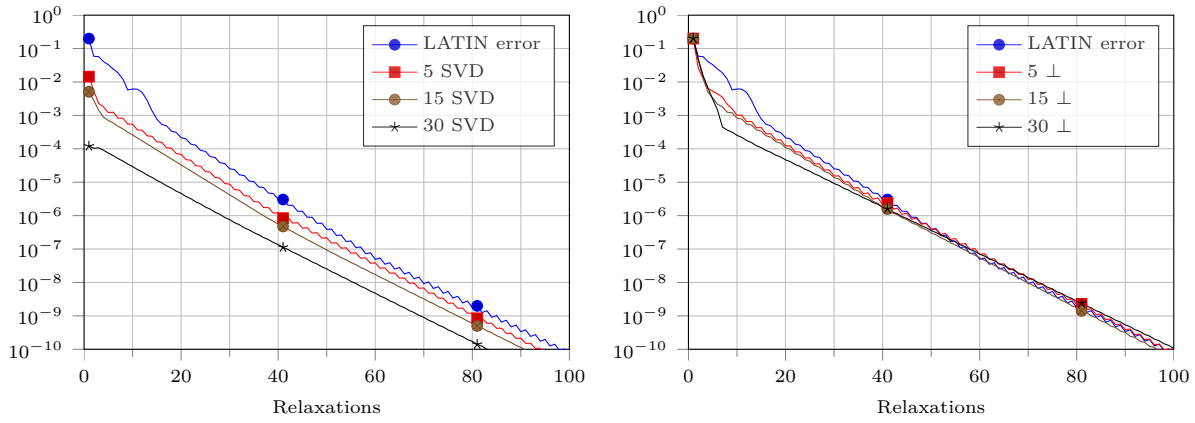
Convergence of the FAS/LATIN method is hereinafter discussed. Its convergence is plotted on figure 2.6 according to the different kinds of space vectors previously described. On those graphs is plotted the evolution of the error versus the number of relaxations (the coarse stage is assumed to be costless).

As a general result, the number of required relaxation stages to reach a given level of accuracy is reduced using the accelerated FAS/LATIN method. The following discussion is strongly bound to the analysis carried out in section 1.5.1 throughout SVD modes of the sought solution. Some important elements are recalled to interpret the FAS/LATIN convergence graphs:

- The reference solution is generated by 31 SVD space-time modes (optimal ones);
- To break the LATIN precision threshold of 10^{-2}, 12 modes are required at least;
- To break the LATIN precision threshold of 10^{-3}, 29 modes are required at least;

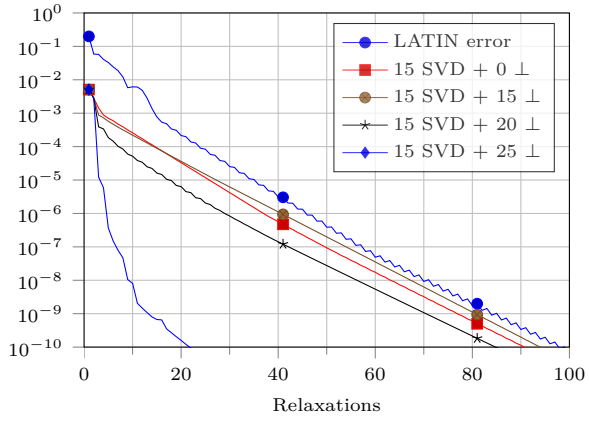
The FAS/LATIN method shows different but interesting behaviors according to the way space vectors are obtained.

On figure 2.6 case (a), the considered space basis is composed of vectors computed on section 1.5.1 with an analysis of the reference solution. The FMG initialization produces an interesting gain when starting the first relaxation stage. Unsurprinsigly, the larger is the basis, the more accurate is the initialization. Moreover, one can remark that the considered space basis is the most suited one to correct the first guess $\mathbf{s}_0 = (\mathbf{u}_0, \boldsymbol{\lambda}_0)$.

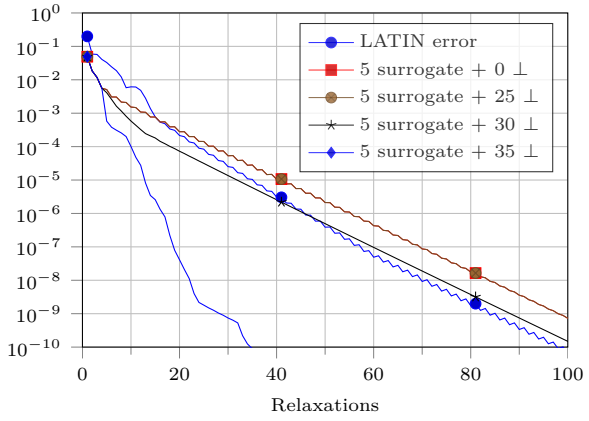


(a) From reference SVD

(b) Computed on-the-fly

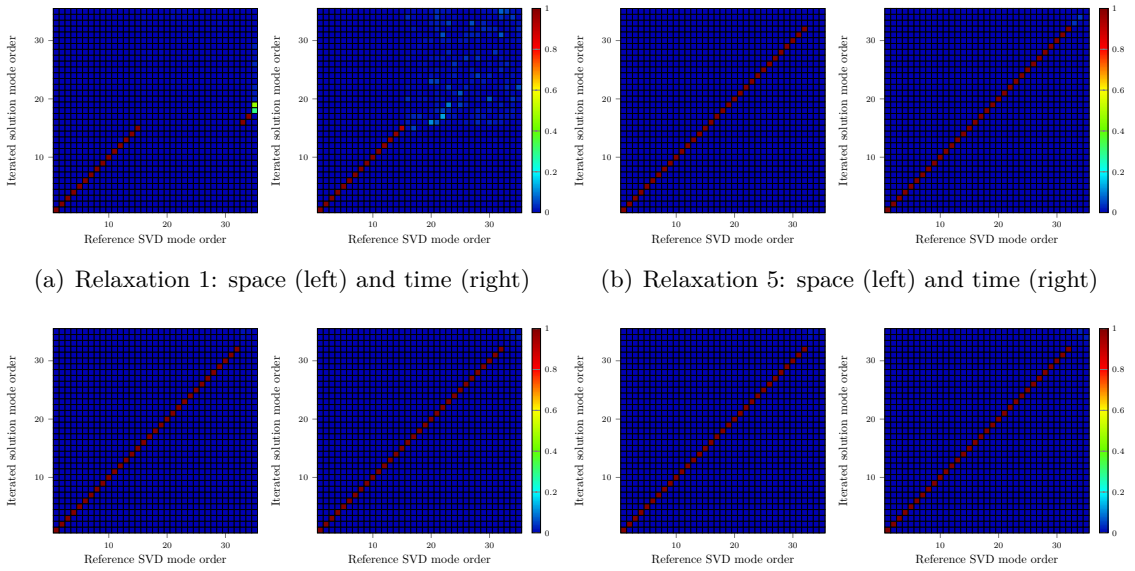


(c) Computed on-the-fly completing SVD ones



(d) From a surrogate problem

Figure 2.6 – FAS/LATIN convergence according different kind of space vectors. Relaxations stage are such that $\nu_1 = \nu_2 = 1$.



(a) Relaxation 1: space (left) and time (right)

(b) Relaxation 5: space (left) and time (right)

(c) Relaxation 10: space (left) and time (right)

(d) Relaxation 15: space (left) and time (right)

Figure 2.7 – MAC diagrams for the extrusion problem solved with the FAS/LATIN method accounting 15 SVD modes and 20 extra vectors computed on the fly.

Moreover, with a 15 space mode basis, the precision reached after the first relaxation stage (roughly 8×10^{-3}) corresponds to the precision of the solution rebuilt from these 15 SVD modes (see figure 1.10). If one has considered the whole SVD space basis, the solution would have been fully computed after the first correction stage. This allows to conclude that the FMG initialization is well-suited. It can be seen that two regimes of convergence occurs. During a first group of relaxations the FAS/LATIN shows a high convergence rate, then the rate of the former LATIN method is recovered. We can suppose that during the first group of relaxations, structural scales are processed and accelerated by the coarse correction stage. Then, no more correction are brought because the considered basis does not allow to compute more accurate solution.

On figure 2.6 case (b), the considered space basis is computed on the fly and updated at each FAS/LATIN cycle according the formula in (2.18). Since the basis of space vectors is initially empty, the FMG initialization is skipped. The FAS/LATIN is highly accelerated during roughly 5 to 8 relaxations according the size of the considered basis. Then, the convergence rate of the LATIN method is recovered. One can remark for instance that in the case of the basis with 30 space modes (which are not ensured to be the optimal ones), the convergence rate modification occurs when the precision 2×10^{-4} is reached. Such a precision corresponds also approximately to the error of the solution rebuilt from the 30 SVD optimal modes. In other words, beyond this precision no more relevant correction can be brought to accelerate the FAS/LATIN convergence rate because the reduced problem is too small.

On figure 2.6 case (c), an interesting compromise seems to be reach by combining the two previous method. The space basis includes 15 space vectors obtained on section 1.5.1 and additional vectors computed on the fly. On the one hand, benefits of the initialization occur as described above. On the other hand, extra vectors produce an accelerated convergence rate during the first relaxation stages. One can remark that to pertain a worthwhile improvement, a total basis size of 35 is required at least. Indeed, accounting the first 30 SVD optimal modes for the solution enables to reach a precision of 4×10^{-4} (see figure 1.10). Consequently, a basis including 30 non optimal vectors cannot provide relevant correction beyond the precision 4×10^{-4}. Remarkably, the total basis size of 40 shows an astonishing accelerated regime and counts a larger number of vectors than necessary to generate the solution. Such a behavior suggests that not only the identified different scales of the solution have to be solved but also other directions generated at relaxation stages have to be corrected. This is achieved by computing extra vectors on-the-fly generating those directions whereon suited corrections are computed. On figure 2.7 are plotted MAC diagrams for the configuration $15 + 20\ \perp$. After the first relaxation, the solution is properly rebuilt according its 15 first SVD modes (thanks to the coarse stage initialization). Then, modes of the solution are quickly found (after only 5 relaxations). Then, the computational work aims to find appropriate amplitudes for found modes and correct iterated but parasitic modes.

Finally, on figure 2.6 case (d), 5 vectors obtained from the surrogate problem instead of the reference one are used and completed with on-the-fly vectors. Only the first 5 space modes (see figure 2.5) are considered in order not to introduce mode characterized by the coarse discretization scale and parasitic influence of interpolation. Of course, the FMG initialization is worse than the one obtained with SVD modes of the solution. But, it produces an interesting start to reach a standard level of accuracy, say 10^{-2}. According to the number of extra vectors considered, the convergence to the precision 10^{-3} is globally faster. Nevertheless, one can remark that beyond this precision, the classic LATIN method could provide a faster convergence rate. Coarse stage may introduce misleading corrections. Using a very large basis, a highly accelerated convergence rate is also gained for the same reason as suggested before.

Throughout the FAS/LATIN and given an appropriate basis, one is able to take advantage of model reduction methods thanks to the global space-time approach. The a posteriori approach shows great abilities to reuse prior computations to accelerate new ones. Such an approach provides an efficient solver to assign parametric studies [GDGR14] for instance. Nevertheless, it has been observed that to pertain an accelerated regime, additional vectors have to be computed on-the-fly in order to maintain the relevancy of the reduced basis and hence the coarse problem. The next goal is of course the full a priori approach solving the problem exclusively into a reduced subspace. This way expensive relation stages on the full dimensional model are avoided. For that purpose, PGD methods seem to be at the forefront. They consists in building progressively an appropriate

reduced subspace (*i.e.* identify the different scales of the problem) wherein the problem is solved. Such techniques are developed in the next chapter.

Chapter 3

Toward an optimal a priori reduced basis strategy

Abstract

The FAS/LATIN method shows limits of pure a posteriori methods. This third chapter suggests a non-linear a priori reduced basis strategy to solve frictional contact problems. For that purpose, a LATIN formulation is proposed with practical developments issued from the proper generalized decomposition. The resulting method exhibits a quasi-optimal property. It is able to compute progressively a frictional solution close to its SVD expansion. This property will be illustrated and discussed throughout numerical applications. Finally, a pre-computation technique for PGD modes from a surrogate model is also suggested.

Contents

3.1 The proper generalized decomposition

Reduced basis method are precious tools to design efficient solvers. Given prior information (*e.g.* prior computations, surrogate models ...), one is able to perform cheap but relevant computations. For instance, the FAS/LATIN method reuses prior knowledge to compute corrections on a reduced basis accelerating the overall convergence of the LATIN method. As it was exemplified in the previous chapter, main drawbacks of the FAS/LATIN method and more generally of a posteriori methods rely on the prior requirements which can be expensive to obtain. Accuracy and effectiveness of the FAS/LATIN depends strongly on the quality of the considered reduced basis and to pertain a worthwhile acceleration, the considered subspace has to be adapted according to the desired precision.

A priori methods seem to be particularly suited to assign this challenge. Contrary to a posteriori methods, not only they do not require prior knowledge about the solution but also, the solution is sought into a reduced subspace designed on-the-fly. The proper generalized decomposition (PGD) is one of the most widespread methods which has demonstrated great abilities to solve efficiently both linear and non-linear problems [CAC10, CKL14, CC14]. Over the 10 last years, this novel technique surges in enthusiasm for its aptitude to tackle high dimensional models in various computational fields (from kinetic theory or quantum chemistry to fluid and solid mechanics [AMCK06, AMCK07]). The PGD is based on the so-called separated representation. It consists in searching a solution, say u, under the following separated form:

$$u(x_1, x_2, ..., x_N) \simeq \check{u}(x_1, x_2, ..., x_N) = \sum_{i=1}^{p} f_i^1(x_1) \times f_i^2(x_2) \times \cdots \times f_i^N(x_N) \tag{3.1}$$

with x_i are generic variables of the problem (*e.g.* space variable, time variable, parameters of the problem, etc) and f_i are univariate functions of the PGD expansion. This representation was first introduced in [Lad99] and called the radial space-time approximation. It was also designed to limit memory usage.

To illustrate the ingenuity of this approach, let us take a pragmatic and imaginary example. A linear and time dependent solid mechanics problem is tackled with the finite element method. For sake of simplicity, the resulting discretized problem is assumed to have N degrees of freedom and N time steps. Thus, the size of the space-time problem is $N \times N$. With the PGD method, one searches the generalized displacement field vector $\mathbf{u}(t)$ into the following separated form:

$$\mathbf{u}(t) \simeq \check{\mathbf{u}}(t) = \sum_{i=1}^{p} \mathbf{V}_i v_i(t) = \sum_{i=1}^{p} \underbrace{\mathbf{V}_i \mathbf{v}_i^T}_{\substack{\text{rank 1 matrix} \\ \text{or dyad}}} \tag{3.2}$$

where $\mathbf{V}_i$ are generalized displacement vectors and $\mathbf{v}_i$ are time vectors both of size N. The above PGD expansion is a low rank approximation. Considering criterion (1.15) for

p, one can expect to reach a reasonable accuracy for $\check{\mathbf{u}}(t)$ with a small expansion size. Then, data to store requires a size of $p \times (N+N)$ instead of $N \times N$. Now, let us increase the dimensionality of the problem by adding an additional parameter, say μ. Then, one has to search the solution $\mathbf{u}(t,\mu)$ for N discrete values for $\mu \in [\mu_1, \cdots, \mu_N]$. Using the brute force approach leads to solve the solution $\mathbf{u}(t,\mu)$ of size $N \times N \times N$ while the PGD method provides the solution under the following low rank tensor approximation:

$$\mathbf{u}(t,\mu) \simeq \check{\mathbf{u}}(t,\mu) = \sum_{i=1}^{p} \underbrace{\mathbf{V}_i \otimes \mathbf{v}_i(t) \otimes \boldsymbol{\nu}_i(\mu)}_{\text{rank 1 tensor}} \tag{3.3}$$

with $\boldsymbol{\nu}_i(\mu)$ is an additional univariate function according to the parameter μ. Then, the amount of data to store becomes $p \times (N+N+N)$. Throughout this example, we understand that the PGD is an exciting method to overcome dimensionality issues for engineering problems. PGD has a complexity scaled linearly according to the space wherein the problem is defined while brute force mesh based methods have a complexity scaled exponentially. Furthermore, PGD methods allows to reduce drastically time of computations by focusing the computational work on univariate functions. In our example, instead of tackling a whole $N \times N \times N$ approximation, only functions of size N have to be handled. By doing so, a great computational work can be spared. PGD approach was also extended to stochastic problems [GLMN14]. A detailed review of several PGD methods is given in [Nou10].

Despite its appeal, PGD remains nowadays challenging especially for parametric analysis. The main difficulty is to provide a practicable low rank expansion (*i.e.* p is small) ensuring a reasonable accuracy for $\check{\mathbf{u}}$. For space-time problems, tensors to handle are second order and handled easily with matrix algebra. Nevertheless for higher order tensors, extension to the tensor algebra is not straightforward at all. Nowadays, tensor decomposition or tensor rank determination remain highly challenging [Kol01, DLDMV00b, DLDMV00a, LDH11]. HO-SVD and PARAFAC methods aim at providing equivalents of the SVD decomposition for high order tensors and are still widely studied.

In this dissertation, the PGD method is focused on. But for both linear and nonlinear problems other a priori methods exist such as the a priori reduction and the a priori hyper-reduction method (APR and APHR) described in [Ryc05, RCCA06].

3.2 Space-time separated form mastering

In the following, a space-time separation is considered to build the PGD expansion of a vector field of interest. For instance, the displacement field $\mathbf{u}$ is sought under the following form:

$$\mathbf{u} \simeq \check{\mathbf{u}} = \mathbf{u}_0 + \mathbf{u}^* = \mathbf{u}_0 + \sum_{i=1}^{p} \mathbf{V}_i \phi_i(t) = \mathbf{u}_0 + \sum_{i=1}^{p} \mathbf{V}_i \boldsymbol{\phi}_i^T \tag{3.4}$$

with the initial guess $\mathbf{u}_0$. For the point of view of memory usage, it can be envisaged that $\mathbf{u}_0$ is obtained also with a PGD method and hence provided into a compressed separated form. The PGD method is applied to build progressively a low rank approximation for $\mathbf{u}^*$. It proceeds by successive enrichments in such a way that accuracy of $\mathbf{u}^*$ improves.

In this section, optimality of expansion (3.4) is discussed. For specific problems (*e.g.* data post-compression), PGD can be used to compute SVD of given snapshots [CKL14]. But for general problems, even if they are expected to be close, PGD modes does not correspond to the optimal ones. In our case and due to the involved non-smooth behaviors, the resolution of frictional contact problems requires to capture precisely tough and brutal status switch (contact / no contact, sliding / sticking, etc.). Even if the solutions show a remarkable "regularity" as well as a strong scale separability throughout its space-time modes, finding them a priori is challenging. To capture them, one can be expected that a large amount of attempts (numerous PGD enrichments) will be required before finding suited ones. Consequently, an efficient strategy has to be implemented in order to identify dominant and contributory modes and get rid of useless iterated ones. More generally, the concept of optimality of the PGD expansion has to be handled. Before running the PGD method, one has to master its expansion for sake of memory usage and computational efficiency. The inflation of the PGD expansion will be illustrated in the following on a numerical application. Hereinafter, we denote respectively by $(\mathbf{V}_k)_1^p$ and $(\phi_k)_1^p$, $p-$tuples of space and time modes (or functions) for a some space-time field. These are respectively a space basis or a time basis for the PGD expansion.

3.2.1 Orthonormality condition for space modes

First, an orthonormality condition is considered for space modes (*i.e.* $\mathbf{V}_i^T\mathbf{V}_j = \delta_{ij}$ with $\boldsymbol{\delta}_{ij}$ the Kronecker symbol). Assuming a new pair $(\mathbf{V}, \phi)$ is provided, it has to complete a preexisting PGD expansion respecting orthonormality condition for space modes. For that purpose, one can use algorithm 6.

Numerical experiments show that orthonormality is not enough to stem PGD expansion inflation as described previously (see the next numerical applications). Indeed, it achieves a compression for space modes but time modes are not tackled. Observations show that an important redundancy occurs for these last.

3.2.2 Toward quasi-optimal space-time separated form

Optimality property for the PGD expansion can be guaranteed by performing an expensive SVD computation of the sought field after each enrichment stage [GR70, BGVD12] (brute force method). A less expansive method consists in deploying SVD updating techniques [BN78, Bra06]. Nevertheless, all of those lie on burdensome methods. Such a high

Algorithm 6: Orthogonal enrichment.

Input:
Preexisting $p-$tuples $(\mathbf{V}_k)_1^p$ and $(\phi_k)_1^p$ with $\mathbf{V}_i^T\mathbf{V}_j = \boldsymbol{\delta}_{ij}$
New pair of modes $\mathbf{V}$ and ϕ
Output:
Enriched $(p+1)$-tuples $(\mathbf{V}_k)_1^{p+1}$ and $(\phi_k)_1^{p+1}$ with $\mathbf{V}_i^T\mathbf{V}_j = \boldsymbol{\delta}_{ij}$

for $k = 1$ **to** p **do**
  Projection on existing space modes $\alpha = \mathbf{V}_k^T\mathbf{V}$
  Update corresponding time modes $\phi_k = \phi_k + \alpha\phi$
  Subtract projected component $\mathbf{V} \leftarrow \mathbf{V} - \alpha\mathbf{V}_k$
end
Norm of remaining space mode $\beta = \|\mathbf{V}\|_2$
Normalize space mode $\mathbf{V}_{p+1} \leftarrow \mathbf{V}/\beta$
Amplify time mode $\phi_{p+1} \leftarrow \beta\phi$

computational effort is not worthwhile knowing that the current iterated field may be inaccurate relatively to the desired level of accuracy. So, enforcing the strict optimality is seemingly useless. Instead, a soft compression or downsizing approach is preferred (*i.e.* diminishing the size p without actually reaching its minimal size but sustaining a good quality for the spanned field). For that purpose, the following space-time projection strategy is implemented. In order to illustrate it, let us take an approximation $\check{\mathbf{u}}$ of rank 2 written as follows:

$$\check{\mathbf{u}} = \mathbf{V}_1\boldsymbol{\phi}_1^T + \mathbf{V}_2\boldsymbol{\phi}_2^T \qquad \text{with} \qquad \mathbf{V}_i\mathbf{V}_j = \delta_{ij} \tag{3.5}$$

Equation (3.5) can be rewritten considering that $\boldsymbol{\phi}_2$ has a redundant component with $\boldsymbol{\phi}_1$ (i.e. $\boldsymbol{\phi}_2 = \alpha\boldsymbol{\phi}_1 + \bar{\boldsymbol{\phi}}_2$ and $\bar{\boldsymbol{\phi}}_2^T\boldsymbol{\phi}_1 = 0$) as follows:

$$\check{\mathbf{u}} = (\mathbf{V}_1 + \alpha\mathbf{V}_2)\boldsymbol{\phi}_1^T + \mathbf{V}_2\bar{\boldsymbol{\phi}}_2^T = \bar{\mathbf{V}}_1\boldsymbol{\phi}_1^T + \mathbf{V}_2\bar{\boldsymbol{\phi}}_2^T \tag{3.6}$$

Rewriting (3.6) does not respect orthogonality for space vectors (i.e. $\mathbf{V}_2 = \beta\bar{\mathbf{V}}_1 + \bar{\mathbf{V}}_2$ and $\bar{\mathbf{V}}_1^T\bar{\mathbf{V}}_2 = 0$). Applying a re-orthogonalization for spatial modes yields to:

$$\check{\mathbf{u}} = \bar{\mathbf{V}}_1(\boldsymbol{\phi}_1^T + \beta\bar{\boldsymbol{\phi}}_2^T) + \bar{\mathbf{V}}_2\bar{\boldsymbol{\phi}}_2^T = \bar{\mathbf{V}}_1\bar{\boldsymbol{\phi}}_1^T + \bar{\mathbf{V}}_2\bar{\boldsymbol{\phi}}_2^T \tag{3.7}$$

Finally, norms of spatial modes can be normalized by reporting amplitudes in time modes as follows

$$\check{\mathbf{u}} = \frac{\bar{\mathbf{V}}_1}{\|\bar{\mathbf{V}}_1\|}\left(\|\bar{\mathbf{V}}_1\|\bar{\boldsymbol{\phi}}_1^T\right) + \frac{\bar{\mathbf{V}}_2}{\|\bar{\mathbf{V}}_2\|}\left(\|\bar{\mathbf{V}}_2\|\bar{\boldsymbol{\phi}}_2^T\right) \tag{3.8}$$

This process does not change $\check{\mathbf{u}}$ and preserves orthogonality property for space modes but does not ensure it for time modes. The first projection step (3.6) compresses redundancy between $\boldsymbol{\phi}_1$ and $\boldsymbol{\phi}_2$ into a new vector $\bar{\boldsymbol{\phi}}_1$ but looses the orthogonality for space

vectors justifying step (3.7). Incidentally, enforcing this property makes $\bar{\boldsymbol{\phi}}_1$ and $\bar{\boldsymbol{\phi}}_2$ partly linearly dependent (but less than original vectors). This process can be repeated until having $\bar{\boldsymbol{\phi}}_1$ and $\bar{\boldsymbol{\phi}}_2$ orthogonal (*i.e.* producing the SVD of $\check{\mathbf{u}}$). Algorithm 7 generalizes the above process for larger rank for $\check{\mathbf{u}}$. Steps (3.6), (3.7) are applied respecting a prescribed sorting (according to time modes amplitudes) for space-time modes. The weakest amplitude space-time dyad is projected on larger amplitude ones. More details about this process is given in appendix A.

Algorithm 7: Downsizing.

Input:
Existing $p-$tuples $(\mathbf{V}_k)_1^p$ and $(\boldsymbol{\phi}_k)_1^p$ with $\mathbf{V}_i^T\mathbf{V}_j = \boldsymbol{\delta}_{ij}$
Max iterations $\xi_{\max}$
Cut-off amplitude ε
Output:
Downsized $q-$tuples $(\mathbf{V}_k)_1^q$ and $(\boldsymbol{\phi}_k)_1^q$ with $\mathbf{V}_i^T\mathbf{V}_j = \boldsymbol{\delta}_{ij}$

```
for ξ = 1 to ξ_max do
    Sort space-time modes such that ‖φ_1‖_2 ⩾ ··· ⩾ ‖φ_p‖_2
    for i = p down to 2 do
        for j = 1 up to i − 1 do
            Projection of time mode α = φ_j^T φ_i/(φ_j^T φ_j)
            Update corresponding space mode V_j ← V_j + αV_i
            Subtract projected component φ_i ← φ_i − αφ_j
            Projection of space mode β = V_j^T V_i/(V_j^T V_j)
            Update corresponding time mode V_i ← V_i − βV_j
            Subtract projected component φ_j ← φ_j + βφ_i
            Norm of the j^th space mode ζ = ‖V_j‖_2
            Normalization V_j ← V_j/ζ and φ_j ← ζφ_j
        end
        Norm of the i^th space mode γ = ‖V_i‖_2
        Normalization V_i ← V_i/γ and φ_i ← γφ_i
    end
end
for i = p down to 1 do
    if ‖v_i‖_2 ⩽ ε then
        Elimination of V_i and φ_i
        Decrement basis size: p ← p − 1
    end
end
q = p
```

The complexity of algorithm 7 is denoted by c (floating operations) and estimated by

$c/\xi_{\max} = n(\frac{5}{2}p^2 - \frac{1}{2}p - 2) + m(\frac{5}{2}p^2 - \frac{3}{2}p - 1) + \frac{3}{2}p^2 - \frac{1}{2}p - 1 = \mathcal{O}(\frac{3}{2}(n+m)p^2)$. A cut-off amplitude ε can be fixed relatively to a desired accuracy for $\check{\mathbf{u}}$. Embedded in the PGD process, the downsizing method stems efficiently the inflation of the PGD expansion. In practice, a very few iterations $\xi_{\max}$ (1 or 2 iterations) is enough. More iterations make converge to the SVD of $\check{\mathbf{u}}$. To compare with classical and widespread techniques: Lanczos thin-SVD method has a $\mathcal{O}(nmp^2)$ complexity [GVL12] and fastest SVD updating methods have a complexity $\mathcal{O}(nmp)$ [Bra02]. All in all, those practical developments provide efficient tools to compress and deal with PGD low rank approximation of the solution considered as quasi-optimal.

3.3 Toward a quasi-optimal a priori reduced basis strategy

In this section, the LATIN method including the space-time separated representation is presented. Such a method coupled with the previous practical developments is able to build a suitable PGD expansion for the solution.

3.3.1 Formulation of the LATIN method with space-time separation

As it is the most expensive stage, space-time separation aims to lighten the computational work of the global stage and reduce memory usage for the solution storage. The local stage remains the same. The LATIN method including the space-time separated representation consists in searching fields $\mathbf{s}^* = (\mathbf{u}^*, \boldsymbol{\lambda}^*)$ into a separated form such that solution field $\mathbf{s} = (\mathbf{u}, \boldsymbol{\lambda})$ reads:

$$\begin{aligned} \mathbf{u}(t) &= \mathbf{u}_0(t) + \mathbf{u}^*(t) = \mathbf{u}_0(t) + \sum_{k=1}^{p} \mathbf{V}_k \phi_k(t) \\ \boldsymbol{\lambda}(t) &= \boldsymbol{\lambda}_0(t) + \boldsymbol{\lambda}^*(t) = \boldsymbol{\lambda}_0(t) + \sum_{k=1}^{q} \mathbf{L}_k \psi_k(t) \end{aligned} \tag{3.9}$$

Vectors $\mathbf{V}_k$ (resp. $\mathbf{L}_k$) are space modes associated to the displacement field defined over Ω (resp. contact force mode defined over $\partial_3\Omega$) and functions $\phi_k(t)$ (resp. $\psi_k(t)$) are time functions associated to $\mathbf{V}_k$ (resp. $\mathbf{L}_k$) defined on $[0, T]$. Vectors $\mathbf{u}_0$ and $\boldsymbol{\lambda}_0$ are initial guesses. The approximation (3.9) can be written equivalently using a matrix notation as follows:

$$\begin{aligned} \mathbf{u} &= \mathbf{u}_0 + \sum_{k=1}^{p} \mathbf{V}_k \boldsymbol{\phi}_k^T \\ \boldsymbol{\lambda} &= \boldsymbol{\lambda}_0 + \sum_{k=1}^{q} \mathbf{L}_k \boldsymbol{\psi}_k^T \end{aligned} \tag{3.10}$$

At the i^{th} iteration, solution $\mathbf{s}_{i-1} = (\mathbf{u}_{i-1}, \boldsymbol{\lambda}_{i-1})$ and the last local stage result $\hat{\mathbf{s}} = (\hat{\mathbf{v}}, \hat{\boldsymbol{\lambda}})$ are given. Considering the correction scheme of the global stage (see section 1.6.2), a

correction $\boldsymbol{\Delta s} = (\boldsymbol{\Delta u}, \boldsymbol{\Delta \lambda})$ is sought such that accuracy of the updated solution $\mathbf{s}_i = \mathbf{s}_{i-1} + \boldsymbol{\Delta s}$ improves. The LATIN formulation without separated space-time representation provides the following system to compute suitable corrections:

$$\begin{cases} \mathbf{K}\boldsymbol{\Delta u} = \mathbf{B}^T \boldsymbol{\Delta \lambda} \\ \boldsymbol{\Delta v} = \mathbf{B}\boldsymbol{\Delta u} \\ \boldsymbol{\Delta \lambda} + \mathbf{k}\boldsymbol{\Delta v} - \mathbf{res}_{\text{sd}} = 0 \end{cases} \tag{3.11}$$

where $\mathbf{res}_{\text{sd}} = \hat{\boldsymbol{\lambda}} - \boldsymbol{\lambda}_{i-1} + \mathbf{k}(\hat{\mathbf{v}} - \mathbf{v}_{i-1})$ is known at this stage. The separated representation is now introduced for corrective increments with space and time modes:

$$\begin{cases} \boldsymbol{\Delta u} = \mathbf{V}\phi(t) & \text{on } \Omega \times [0,T] \\ \boldsymbol{\Delta \lambda} = \mathbf{L}\psi(t) & \text{on } \partial_3\Omega \times [0,T] \end{cases} \tag{3.12}$$

Accounting space-time representation (3.12) in addition to linear system (3.11) makes the global stage overdetermined. Then, solution is computed such that $\mathbf{s} \in \mathcal{A}$ (*i.e.* internal balance equation is respected) as well as space-time separated representation. But search direction equations are approximated (*i.e.* verified at best or in a weak sense). Substituting representations (3.12) into internal balance equation leads to the following conditions for sought modes:

$$\forall t \in [0,T] : \ \mathbf{K}\mathbf{V}\phi(t) = \mathbf{B}^T\mathbf{L}\psi(t) \quad \Leftrightarrow \quad \begin{cases} \mathbf{KV} = \mathbf{B}^T\mathbf{L} \\ \phi(t) = \psi(t) \end{cases} \tag{3.13}$$

All in all to solve the global stage with a space-time representation, the above conditions have to be respected. By doing so, search direction equation is verified at best by solving the following optimization problem in the general case:

$$\{\mathbf{W}, \phi(t)\} = \underset{\mathbf{W}^\star, \phi^\star(t)}{\operatorname{argmin}} \|\mathbf{W}^\star\phi^\star(t) - \mathbf{res}_{\text{sd}}\|_F \tag{3.14}$$

with $\mathbf{W} = \mathbf{L} + \mathbf{kBV}$. $\mathbf{W}$ and $\phi(t)$ have to be found in such a way that (3.14) is verified at best. Definition of space mode $\mathbf{W}$ is a computational trick. Indeed, to avoid complex algebraic manipulations with equations (constraint (3.13) to substitute into (3.14)), an auxiliary space mode $\mathbf{W}$ is defined arousing an easy formulation for the optimization problem (3.14).

Once $\mathbf{W}$ and $\phi(t)$ are found, the global stage ends. Those vectors corresponds to enrichments for the on-going solution which includes preexistent represented corrections. In practice, only modes $\mathbf{W}$ and time modes $\phi(t)$ are stored into respective $p-$tuples $(\mathbf{W}_k)_1^p$ and $(\phi_k)_1^p$ accounting the orthonormality and size control. When it is required (*e.g.* one wants to build solution fields), associated vectors $(\mathbf{V}_k)_1^p$ and $(\mathbf{L}_k)_1^p$ can be deduced,

respecting conditions (3.13), as follows:

$$\forall k \in [\![0,p]\!] : \begin{cases} [\mathbf{K} + \mathbf{B}^T\mathbf{k}\mathbf{B}]\mathbf{V}_k = \mathbf{B}^T\mathbf{W}_k \\ \mathbf{L}_k = \mathbf{W}_k - \mathbf{k}\mathbf{B}\mathbf{V}_k \\ \psi_k = \phi_k \end{cases} \tag{3.15}$$

and then solution fields rebuild according to (3.9). moreover, at a given iteration, the quantity $\mathbf{res}_{\text{sd}}$ can be computed directly from $(\mathbf{W}_k)_1^p$ and $(\phi_k)_1^p$ as follows:

$$\mathbf{res}_{\text{sd}} = (\boldsymbol{\lambda}_0 + \mathbf{k}\mathbf{B}\mathbf{u}_0) - (\hat{\boldsymbol{\lambda}} + \mathbf{k}\hat{\mathbf{u}}) + \sum_{k=1}^{p} \mathbf{W}_k\phi_k(t) \tag{3.16}$$

To solve the optimization problem (3.14), several approaches could be used.

- Time function updating (preliminary stage).
- Space function updating.
- Enrichment stage.

3.3.2 Preliminary stage

One looks for solving global stage defined in (3.14) with prescribed space modes $(\mathbf{W}_k)_1^p$ and a corresponding set of time functions $(\widetilde{\phi}_k)_1^p$ is sought. Time modes $(\widetilde{\phi}_k)_1^p$ are solution of the following optimization problem:

$$\widetilde{\phi}_k = \underset{\phi^\star}{\text{argmin}} \, \|\mathbf{W}_k\phi^\star - \mathbf{res}_{\text{sd}}\|_F \tag{3.17}$$

To solve the above problem, a Galerkin method is applied with the trial function $\boldsymbol{\omega}^\star = \mathbf{W}_k\phi^\star(t)$ and yields to:

$$\widetilde{\phi}_k = \mathbf{W}_k^T\mathbf{res}_{\text{sd}}/(\mathbf{W}_k^T\mathbf{W}_k) \tag{3.18}$$

Then, updated set of time modes are $(\phi_k)_1^p \leftarrow (\phi_k + \widetilde{\phi}_k)_1^p$. The corresponding pseudo-code of this approach is given in algorithm 8.

Thanks to orthonormality property for $(\mathbf{W}_k)_1^p$, quantity $\mathbf{res}_{\text{sd}}$ has not not to be reevaluated after each time function update. Hence, this approach is highly parallelisable.

3.3.3 Extension of the preliminary stage

An extension of this preliminary phase is proposed by updating also space modes. In other words, global stage (3.14) is solved with prescribed time modes $(\phi_k)_1^p$. Then, new corresponding $(\widetilde{\mathbf{W}}_k)_1^p$ space modes have to be found according to the following optimization problem:

$$\widetilde{\mathbf{W}}_k = \underset{\mathbf{W}^\star}{\text{argmin}} \, \|\mathbf{W}^\star\phi_k - \mathbf{res}_{\text{sd}}\|_2 \tag{3.19}$$

Algorithm 8: Time modes update.

Input:
$\mathbf{res}_{\mathrm{sd}}$
p-tuple of existing space modes $(\mathbf{W}_k)_1^p$
p-tuple of existing time modes $(\phi_k)_1^p$
Output:
p-tuple of updated time modes $(\phi_k)_1^p$

```
for k = 1 to p do
    for t_{1⩽k⩽m} do
        Update time modes φ_k(t_k) ← φ_k(t_k) + W_k^T res_sd(t_k)/(W_k^T W_k)
    end
end
```

Again, a Galerkin method with the trial function $\boldsymbol{\omega}^\star = \mathbf{W}^\star \phi_k(t)$ is applied and yields to:

$$\widetilde{\mathbf{W}}_k = \int_0^T \mathbf{res}_{\mathrm{sd}} \phi_k(t)\,\mathrm{d}t \Big/ \int_0^T \phi_k^2(t)\,\mathrm{d}t \tag{3.20}$$

Because of orthonormality condition, new space modes $(\mathbf{W}_k)_1^p$ can not be updated directly. Each of them has to be projected in the preexistent spatial basis. The corresponding pseudo-code is given in algorithm 9.

The updating stage for time functions and the updating stage for space functions form an extended preliminary stage. Once it is solved, PGD expansion is modified and becomes more accurate pertaining its size. Nevertheless, one has to keep in mind that the size of a given low rank approximation (even if it is the optimal one) gives a lower-bound for the error. To break through it, one has to extend the PGD basis (*i.e.* enrich the PGD expansion).

3.3.4 Enrichment stage

In [BBL90], the updating stage for time functions as described previously in section 3.3.2 is called the preliminary stage. Then, the enrichment stage is used only if the convergence criterion is not sufficiently decreased by the updating stage (e.g. criterion decreases by a factor less than 10%). The enrichment stage consists in generating a new pair of modes extending the PGD basis. For that purpose, the global stage (3.14) is equivalent to the following optimization problem:

$$\{\mathbf{W}, \phi(t)\} = \underset{\mathbf{W}^\star, \phi^\star(t)}{\operatorname{argmin}} \|\mathbf{W}^\star \phi^\star(t) - \mathbf{res}_{\mathrm{sd}}\|_F \tag{3.21}$$

Algorithm 9: Space modes update.

Input:
$\mathbf{res}_{\mathrm{sd}}$
p-tuple of existing space modes $(\mathbf{W}_k)_1^p$
p-tuple of existing time modes $(\phi_k)_1^p$
Output:
p-tuple of updated space modes $(\mathbf{W}_k)_1^p$

for $i = 1$ **to** p **do**
  New space mode $\widetilde{\mathbf{W}}_i = \int_0^T \mathbf{res}_{\mathrm{sd}} \phi_i(t)\,\mathrm{d}t / \int_0^T \phi_i^2(t)\,\mathrm{d}t$
  Update $\mathbf{res}_{\mathrm{sd}} \leftarrow \mathbf{res}_{\mathrm{sd}} + \widetilde{\mathbf{W}}_i \phi_i(t)$
  for $j = 1$ **to** p **do**
    Projection of the new space mode $\alpha = \mathbf{W}_j^T \widetilde{\mathbf{W}}_i$
    Update corresponding time modes $\phi_j \leftarrow \phi_j + \alpha \phi_i$
    Subtract projected component $\widetilde{\mathbf{W}}_i \leftarrow \widetilde{\mathbf{W}}_i - \alpha \mathbf{W}_j$
  end
  Correction of space mode $\mathbf{W}_i \leftarrow \mathbf{W}_i + \widetilde{\mathbf{W}}_i$
  Norm of space mode $\beta = \|\mathbf{W}_i\|_2$
  Normalization $\mathbf{W}_i \leftarrow \mathbf{W}_i / \beta$
  Amplification $\phi_i \leftarrow \beta \phi_i$
end

Based on a space-time weak form modeling method (1.28), a Galerkin method is applied with the trial function $\boldsymbol{\omega}^\star = \mathbf{W}^\star\phi(t) + \mathbf{W}\phi^\star(t)$ and yields to:

$$\forall(\mathbf{W}^\star,\phi^\star) : \int_0^T \boldsymbol{\omega}^{\star T}\Big(\mathbf{W}\phi(t) - \mathbf{res}_{\text{sd}}\Big)\,\mathrm{d}t = \mathbf{W}^{\star T}\Big(\mathbf{W}\int_0^T \phi^2(t)\,\mathrm{d}t - \int_0^T \mathbf{res}_{\text{sd}}\phi(t)\,\mathrm{d}t\Big) + \int_0^T \phi^\star(t)\Big(\mathbf{W}\phi(t) - \mathbf{res}_{\text{sd}}\Big)\,\mathrm{d}t = 0 \quad (3.22)$$

The non-linear system (3.22) is solved with a robust iterative strategy [CALK11]. It proceeds in two steps, repeated until convergence. The first step consists in computing vector $\mathbf{W}$ knowing $\phi(t)$ from the previous step. Then, the second step consists in updating $\phi(t)$ knowing $\mathbf{W}$ from the first step. The process is stopped when $\mathbf{W}$ and $\phi(t)$ are no more significantly updated.

First step Space vector $\mathbf{W}$ is updated knowing $\phi(t)$ from the previous step:

$$\mathbf{W} = \Big(\int_0^T \mathbf{res}_{\text{sd}}\phi(t)\,\mathrm{d}t\Big)\Big/\Big(\int_0^T \phi^2(t)\,\mathrm{d}t\Big) \quad (3.23)$$

Second step Time vector $\phi(t)$ is updated knowing $\mathbf{W}$ from the first step:

$$\phi(t) = (\mathbf{W}^T\mathbf{res}_{\text{sd}})/(\mathbf{W}^T\mathbf{W}) \quad (3.24)$$

In practice, very few iterations are needed and a normalization condition is applied for one of the two vectors to ensure uniqueness of the product of the two modes. The pseudo-code of this method is given in the algorithm 10.

Algorithm 10: Enrichment stage.

Input:
$\mathbf{res}_{\text{sd}}$
Maximum number of iterations $\xi_{\max}$
Output:
New space and time modes $\mathbf{W}$, $\phi(t)$

Initialize $\phi(t)$ (*e.g.* $\phi(t) = 1$)
for $\xi = 1$ **to** $\xi_{\max}$ **do**
  Compute space mode $\mathbf{W} = (\int_0^T \mathbf{res}_{\text{sd}}\phi(t)\,\mathrm{d}t)/(\int_0^T \phi^2(t)\,\mathrm{d}t)$
  Compute time mode $\phi(t) = (\mathbf{W}^T\mathbf{res}_{\text{sd}})/(\mathbf{W}^T\mathbf{W})$
  Space mode amplitude $\alpha = \|\mathbf{W}\|_2$
  Normalize space mode $\mathbf{W} \leftarrow \mathbf{W}/\alpha$
  Amplify time mode $\phi \leftarrow \alpha\phi$
end

Once the new pair is found, space and time basis are enriched using algorithm 6 (enrichment of $p-$tupes respecting orthonormality condition for space modes).

3.3.5 The quasi-optimal LATIN-PGD algorithm for frictional contact

The whole pseudo-code of the quasi-optimal LATIN-PGD is given in algorithm 11. Note that, the cut-off amplitude criterion of the downsizing stage is choosen in relation to the LATIN-PGD convergence indicator.

3.4 Toward an hybrid a posteriori/a priori strategy

The suggested LATIN-PGD method is an a priori method solving a given frictional contact problem into a suited subspace designed through on-the-fly computed vector basis. Due to the low rank approximation format for the solution arousing cheap computations and low memory usage, the computational work is reduced. Instead of starting from scratch, PGD basis could also be initialized with prior knowledge. For instance, a surrogate model based on coarser discretizations could capture global modes. Given these modes, they can be reused (probably modified and enhanced through PGD process) to start the suggested algorithm.

A surrogate model is assumed to be solved. Then, its solution $\tilde{\mathbf{s}} = (\tilde{\mathbf{u}}, \tilde{\boldsymbol{\lambda}}) = (\tilde{\mathbf{u}}_0, \tilde{\boldsymbol{\lambda}}_0) + (\tilde{\mathbf{u}}^*, \tilde{\boldsymbol{\lambda}}^*)$ is provided. Modes $(\tilde{\mathbf{V}}_k)_1^p$ and modes $(\tilde{\mathbf{L}}_k)_1^p$ spanning $(\mathbf{u}^*, \boldsymbol{\lambda}^*)$ can be deduced with a SVD computation as in section 1.5. Consequently, auxiliary modes $(\tilde{\mathbf{W}}_k)_1^p$ and time modes $(\tilde{\phi}_k)_1^p$ can also be computed. These can be obtained directly by performing the following computation:

$$\tilde{\mathbf{W}}\tilde{\boldsymbol{\Sigma}}\tilde{\boldsymbol{\Phi}}^T = \mathrm{svd}(\tilde{\boldsymbol{\lambda}}^* + \mathbf{k}\mathbf{B}\tilde{\mathbf{u}}^*) \tag{3.25}$$

$(\tilde{\mathbf{W}}_k)_1^p$ and $(\tilde{\phi}_k)_1^p$ are respectively deduced from the p first columns of $\tilde{\mathbf{W}}$ and $(\tilde{\boldsymbol{\Sigma}}\tilde{\boldsymbol{\Phi}}^T)^T$. Next, the LATIN-PGD algorithm is initialized with them instead of empty sets. By doing so, one can expect to represent structural modes and make PGD focused on local modes saving up much more computational time.

3.5 Applications and examples

3.5.1 Extrusion problem

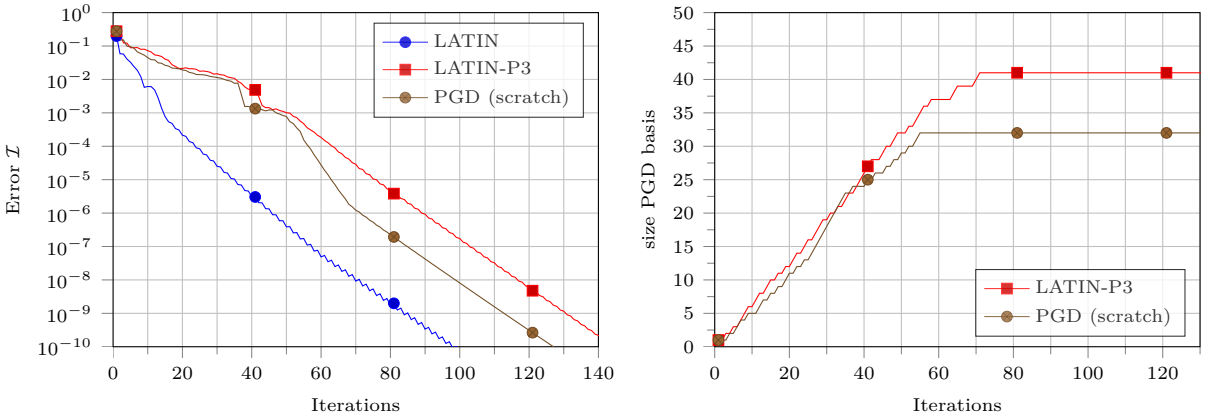
In this section, the suggested LATIN-PGD method is applied to the illustrative example described on section 1.5.1. The convergence is discussed with the evolution of the error and the evolution of the size of the PGD basis $(\mathbf{W}_k)_1^p$ along iterations.

Three different methods are compared on figure 3.1:

Algorithm 11: Quasi-optimal LATIN-PGD algorithm for frictional contact.

Data:
Precisions ε, $\varepsilon_{\max}$
Enriching criterion $0 \leqslant \theta \leqslant 1$

Solve the linear elastic solution $\mathbf{K}\mathbf{u}_0 = \mathbf{f}_{\text{ext}}$ and $\boldsymbol{\lambda}_0 = \mathbf{0}$
Initialization of the PGD basis $p = 0$, $(\mathbf{W}_k)_1^p = \emptyset$ and $(\phi_k)_1^p = \emptyset$
while $\mathcal{I} > \varepsilon$ **do**
    `** Local stage`
    Build solution at the contacting interface $\rightarrow (\mathbf{v}, \boldsymbol{\lambda})$
    Solve the local stage such that $\mathcal{R}(\hat{\mathbf{v}}, \hat{\boldsymbol{\lambda}}) = 0 \rightarrow (\hat{\mathbf{v}}, \hat{\boldsymbol{\lambda}})$
    `** Extended preliminary stage`
    if $p \geqslant 1$ **then**
        for $t_{1 \leqslant k \leqslant m}$ **do**
            Compute $\mathbf{res}_{\text{sd}}(t_k) = (\boldsymbol{\lambda}_0(t_k) + \mathbf{k}\mathbf{B}\mathbf{u}_0(t_k)) - (\hat{\boldsymbol{\lambda}}(t_k) + \mathbf{k}\hat{\mathbf{u}}(t_k)) + \sum_{k=1}^{p} \mathbf{W}_k \phi_k(t_k)$
        end
        Time modes update $\rightarrow (\phi_k)_1^p$
        for $t_{1 \leqslant k \leqslant m}$ **do**
            Compute $\mathbf{res}_{\text{sd}}(t_k) = (\boldsymbol{\lambda}_0(t_k) + \mathbf{k}\mathbf{B}\mathbf{u}_0(t_k)) - (\hat{\boldsymbol{\lambda}}(t_k) + \mathbf{k}\hat{\mathbf{u}}(t_k)) + \sum_{k=1}^{p} \mathbf{W}_k \phi_k(t_k)$
        end
        Space modes update $\rightarrow (\mathbf{W}_k)_1^p$
    end
    `** Enrichment stage`
    if $(1 - \mathcal{I}/\mathcal{I}^{\text{old}} < \theta)$ **then**
        for $t_{1 \leqslant k \leqslant m}$ **do**
            Compute $\mathbf{res}_{\text{sd}}(t_k) = (\boldsymbol{\lambda}_0(t_k) + \mathbf{k}\mathbf{B}\mathbf{u}_0(t_k)) - (\hat{\boldsymbol{\lambda}}(t_k) + \mathbf{k}\hat{\mathbf{u}}(t_k)) + \sum_{k=1}^{p} \mathbf{W}_k \phi_k(t_k)$
        end
        Enrichment stage $\rightarrow \{\mathbf{W}, \phi\}$
        Orthogonal enrichment $\rightarrow \{(\mathbf{W}_k)_1^{p+1}, (\phi_k)_1^{p+1}\}$
        Increment basis size $p \leftarrow p + 1$
    end
    `** Downsizing stage`
    if $p \geqslant 1$ **then**
        Downsizing stage with $\xi_{\max} = 1$ and $\varepsilon_{\max} \rightarrow \{(\mathbf{W}_k)_1^q, (\phi_k)_1^q\}$
        Update basis size $p = q$
    end
    `** Convergence check`
    Save previous convergence criterion $\mathcal{I}^{\text{old}} \leftarrow \mathcal{I}$
    Compute convergence criterion $\mathcal{I}$
end



(a) Convergence of LATIN, LATIN-P3 and PGD

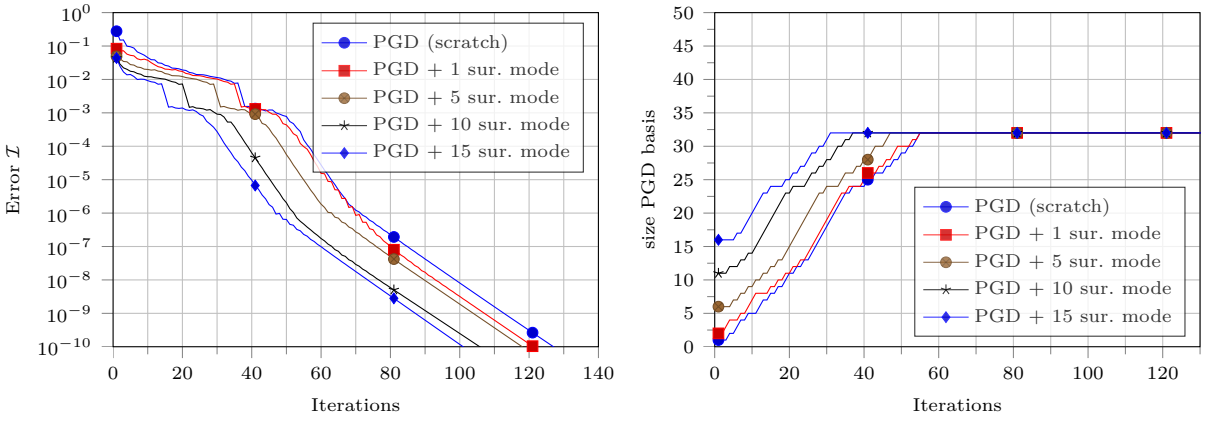
(b) Basis size with LATIN-P3 and PGD

Figure 3.1 – Comparison between the LATIN method, the LATIN method accounting (P3) and the suggested quasi-optimal LATIN-PGD.

- The LATIN method without separated representation considered as the reference non-linear solver. Such a solver solves the problem on the full dimensional model and provides the reference solution;
- The LATIN method as originally proposed in [Lad99] including the space-time representation. For that purpose, the suggested formulation based on $(\mathbf{W}_k)_1^p$ basis is considered accounting the thin preliminary phase;
- The quasi-optimal LATIN-PGD method including the suggested formulation based on $(\mathbf{W}_k)_1^p$, the extended preliminary phase and the downsizing stage.

Remarkably, reduced basis methods (*i.e.* both LATIN-P3 and LATIN-PGD) are able to compute highly accurate solutions. Both converge to the reference solution. This suggests clearly that the working subspace is adapted progressively through iterations and fit iterated solutions according to their level of accuracy. Nevertheless to reach a same level of precision, more iterations are required in comparison to the full dimensional LATIN method. But one has to keep in mind that iterations of LATIN-P3 or LATIN-PGD is far less expensive.

The LATIN-PGD seems to perform better than the LATIN-P3. During the first 50 iterations, both performances and low rank approximation sizes are similar. Note that after 50 iterations, roughly 32 modes are found by both of methods. That corresponds to the size of the subspace spanning the reference solution. Between 50 and 70 iterations, the LATIN-PGD has an accelerated regime until recovering the convergence rate of the LATIN method whereas the LATIN-P3 after 50 iterations. One order of magnitude is gained for the LATIN-PGD. Another interesting observation concerns also basis sizes. The LATIN-PGD does not largely oversize the working subspace to span the solution



(a) Convergence

(b) Basis

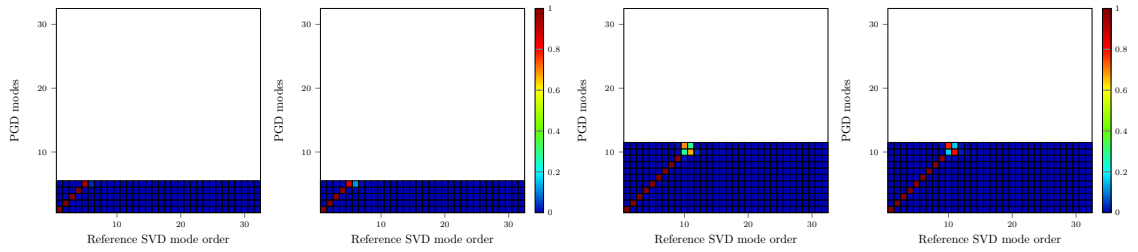
Figure 3.2 – LATIN-PGD with PGD basis initialized with modes from the surrogate model (coarsly discretized).

contrary to the LATIN-P3. The LATIN-PGD computes a basis growing until reaching the required size for the solution and useless additional dimensions are removed thanks to the downsizing stage. On this illustrative case, the PGD basis inflation aroused first does not occur and the orthogonality property considered for the LATIN-P3 is seemingly sufficient to pertain a sustainable working subspace.

On figure 3.2, the PGD basis of the LATIN-PGD is initialized with modes obtained from the surrogate model with coarse discretizations. Initialization with modes obtained from a surrogate model shows auspicious performances. First, these modes provide an enhanced initial guess to start the LATIN-PGD. Indeed, the larger is the number of modes, the better becomes the initial guess. But, one can see that considering more than 5 surrogate modes does not allow to improve the initialization. Indeed, beyond the fifth order, modes are clearly bound to the discretization scale of the surrogate model and does not bring additional relevant information. Regardless to this initial gain, the more modes are considered, the faster is the global convergence (convergence curves are shifted towards lower error values). This suggests that modes initially irrelevant are quickly corrected to span a suited subspace for the considered problem. The last interesting observation is the evolution of the PGD basis size. Starting from an initial basis, the LATIN-PGD algorithm completes and adapts it until it finds the suited basis for the solution (sizing 32) thanks to the downsizing stage.

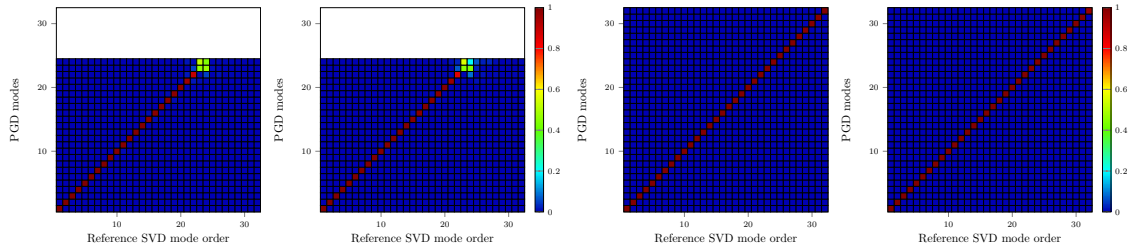
The progressive suitability of the PGD basis through iterations is the so-called quasi-optimality property and an analysis with MAC diagrams is carried out. Starting from scratch, the LATIN-PGD computes on-the-fly PGD modes are compared to reference ones obtained from the solution of the problem (see figure 3.3).

Figure 3.3 shows clearly that iterated PGD modes are close to the reference ones



(a) Iteration 10: space (left) and time (right)

(b) Iteration 20: space (left) and time (right)



(c) Iteration 40: space (left) and time (right)

(d) Iteration 60: space (left) and time (right)

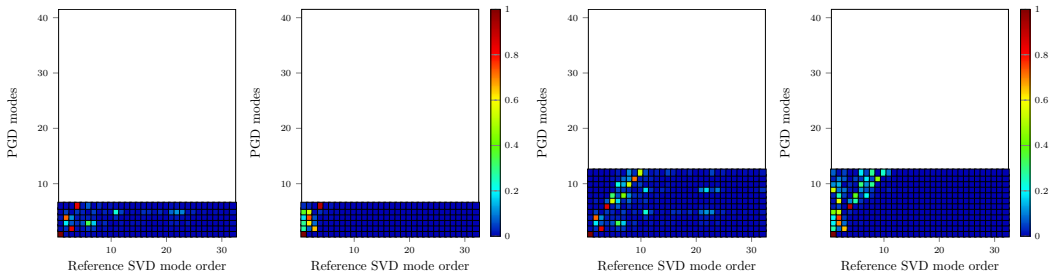
Figure 3.3 – MAC diagrams after given numbers of iteration of LATIN-PGD. Reference modes are computed from the solution of the problem.

on the contrary to modes iterated by the LATIN-P3 method (see figure 3.4). It means that the LATIN-PGD succeeds in capturing quickly the different scales of the problem by identifying modes really close to optimal ones. As it was previously mentioned, SVD modes of the solution depict specific scales of the problem. The same behavior can be illustrated for modes $(\mathbf{W}_k)_1^p$ from which $(\mathbf{V}_k)_1^p$ and $(\mathbf{L}_k)_1^p$ are deduced. All in all, the suggested method computes through iterations a low rank approximation of the solution quite close to the optimal ones, hence achieving a quasi-optimal a priori reduced basis approach.

3.5.2 Three dimensional multibody problem

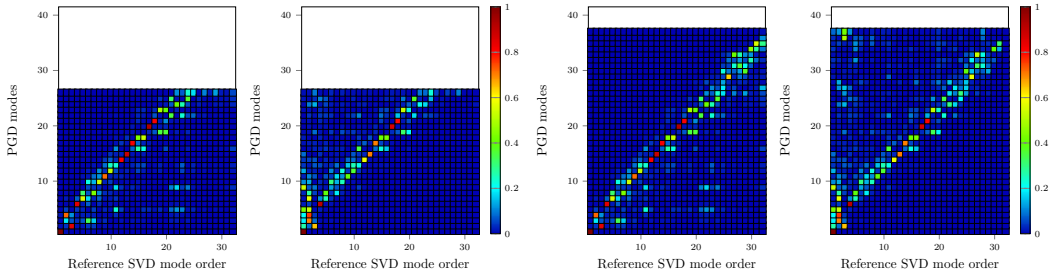
The suggested LATIN-PGD method is also applied successfully to the three dimensional multibody problem described on section 1.5.5. The LATIN method, the LATIN-P3 and the LATIN-PGD are compared on figure 3.5.

For this problem, the LATIN-PGD converges even faster than the LATIN method and the computed PGD expansion has the same behavior as described the previous example: PGD basis grows until reaching roughly the size of the subspace spanning the solution (here, 83 vs. 81 required modes; more sweeps for the downsizing stage may reduce this size to fit more precisely the targeted size 81). The LATIN-P3 shows slightly worse performances than the LATIN-PGD up to the precision 10^{-1} (10 iterations). After this point, its basis inflates greatly despite the prescribed orthonormality and lacks in both



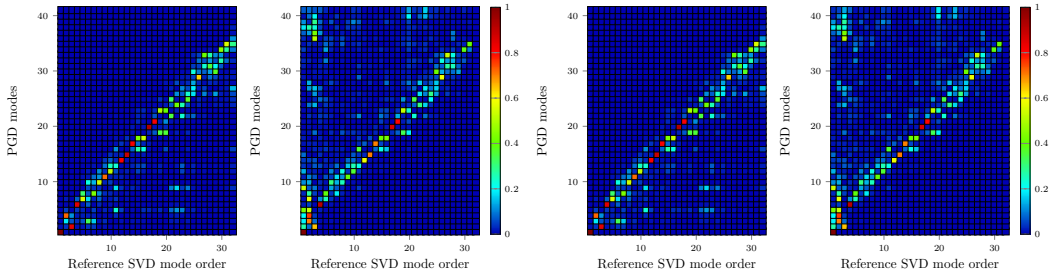
(a) Iteration 10: space (left) and time (right)

(b) Iteration 20: space (left) and time (right)



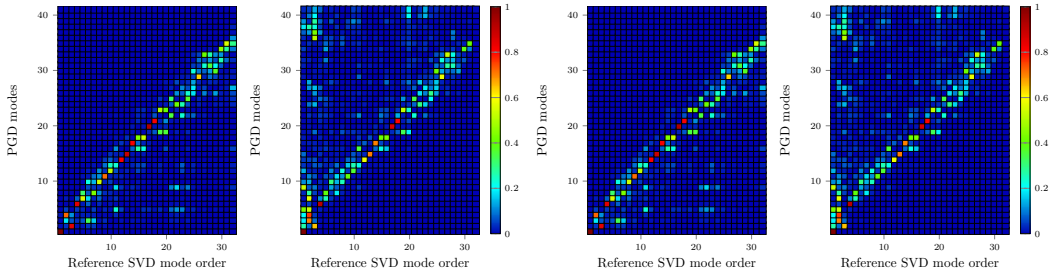
(c) Iteration 40: space (left) and time (right)

(d) Iteration 60: space (left) and time (right)



(e) Iteration 80: space (left) and time (right)

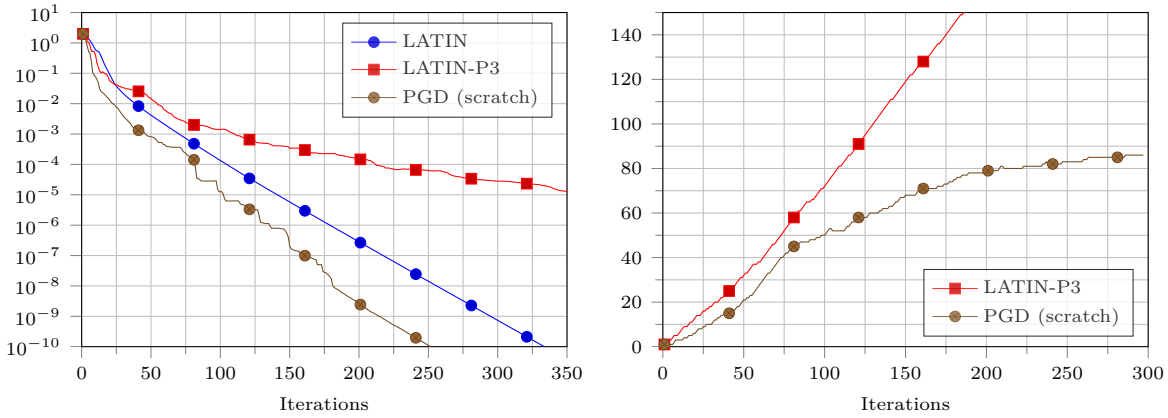
(f) Iteration 100: space (left) and time (right)



(g) Iteration 120: space (left) and time (right)

(h) Iteration 140: space (left) and time (right)

Figure 3.4 – MAC diagrams after given numbers of iteration of LATIN-P3. Reference modes are computed from the solution of the problem.



(a) Convergence of LATIN, LATIN-P3 and PGD

(b) Basis size with LATIN-P3 and PGD

Figure 3.5 – Comparison between the LATIN method, the LATIN method accounting (P3) and the suggested quasi-optimal LATIN-PGD.

convergence rate and computational efficiency (after 350 iterations, LATIN-P3 counts 657 space-time modes !). This example demonstrates particularly the efficiency of the suggested downsizing stage.

For multibody problems, several PGD formulations could be tackled:

- All bodies are tackled globally. The global stage is solved using the PGD formulation. An only pair of space and time modes is computed $(\mathbf{W}_k)_1^p$, $(\phi_k)_1^p$ wherein the whole unknowns are assembled. This approach was here chosen for a sake of simplicity.

- Each body is tackled individually. The global stage is solved using the PGD formulation and pairs of space and time modes are computed $(\mathbf{W}_k^{(1)})_1^{p_1}$, $(\phi_k^{(1)})_1^{p_1}$, $(\mathbf{W}_k^{(2)})_1^{p_2}$, $(\phi_k^{(2)})_1^{p_2}$ ... and are associated to each solid. This approach is suited for parallel computing. But it is harder to handle because multiple sets of vectors have to be handled, and a particular gluing at the contacting interface should be dealt with. Then, domain decomposition methods are useful tools to deal with this issue [LT94, LTS95, BHL98, Hil00, ARLF04, DHK+05, AF09].

Chapter 4

Prospects and future developments

Abstract

The non-linear LATIN solver and PGD method open up several prospects for the suggested strategies. In previous chapters, reductions methods were essentially used to develop efficient solvers for frictional contact problems. Nonetheless, they can also assign efficiently parametric studies. In this last chapter, this aspect will be first investigated and some development paths will be given for both a posteriori reduction methods with ROM and PGD method. Then, the non-linear LATIN method formulation for frictional contact problems will be extended to material non-linearity involving plasticity. A primer a priori reduction method for this extended non-linear solver will be also given.

Contents

4.1 Extensions to parametric studies and real time computing

Usually, to design a mechanical structure, engineers aim to meet the best compromise between performance and cost. Such an objective involves most of the time to choose among several possibilities, for instance material properties and physical parameters. Simulation tools are then helpful to ease the decision-making process. Nonetheless, wide ranges for parameters lead to carry out numerous computations resulting in large computational times. Up to this chapter, reduction methods were exclusively used to design efficient solvers. Nevertheless, they are also well suited to assign parametric studies. Parameters to handle are generally related to (i) the geometry of studied bodies or to (ii) the material properties and possibly boundary conditions:

(i) **Geometry**. For shape optimization design, specific reduced basis approaches have to be developed as suggested in [RHP08, LR10, RM10, RLM11]. To study the influence of given parameters on a varying geometry, projection methods have to be considered [BM00, GJMN05, HL03, DB06] or mesh morphing techniques [Ale02].

(ii) **Material properties**. As the studied space-time domain remains the same, implementation of reduction methods is easier in this case. It consists in performing some computations for some well chosen parameter values (also called design points). Then, a reduced basis is deduced from these snapshots and it is expected to be relevant for the whole parameter sets. Finally, remaining computations are carried out on the deduced ROM. The main challenge consists in choosing appropriate design points resulting in a relevant ROM [HBN13, AFL07, ACF10, AZF12a]. The PGD could also be extended to parametric studies. This a priori approach consists in finding a whole space-time-parameter solution into a low rank tensor approximation and it is expected to be cheap. Then, solutions associated to different parameter values are post-processed (*i.e.* no online phase is required) as in [NACC08, CLB+13]. The challenge lies essentially on the computation of a low rank tensor approximation format. As suggested in the previous chapter, mastering high order tensor is actually not straightforward.

Several other tools such as response surface methods may also be used to assign parametric and optimization problems. Advantages of such reduced basis strategy lies on physically-based formulations contrary to pure statistical methods. Consequently, resulting approximated models are generally more relevant.

Suggested FAS/LATIN method and LATIN-PGD method can also handle parametric studies [GDGR14] by considering as surrogate models some design point. Then, a basis reusing strategy as in [RCB11] can be deployed to solved accurately each configuration in the parametric space. In the following, some development paths are given to address parametric studies related to material properties. Previously discussed methods are reviewed or extended to be parametric-oriented.

4.1.1 The LATIN-ROM solver

In this section, a non-linear ROM based on the LATIN method formulated for frictional contact is suggested. Actually, it corresponds to the coarse problem of the FAS/LATIN. Given a reduced basis (stored in the operator $\mathbf{P}$), the coarse problem computes very cheaply the most suited solution belonging to the spanned subspace satisfying at best frictional contact conditions. The expensive global stage of the LATIN method is substituted with the reduced global stage as described in section 2.2.2 without accounting the FAS correction scheme. The corresponding pseudo-code of the LATIN-ROM is described in algorithm 12.

Algorithm 12: The LATIN-ROM method with prescribed space modes.

Input:
First guess of the solution $\mathbf{s}_0 = (\mathbf{u}_0, \boldsymbol{\lambda}_0)$
Precision ε
Maximum number of LATIN-ROM iterations $k_{\max}$
Matrix of vector basis $\mathbf{P}$
Stagnation criterion θ

Initialization $k = 0$
Compute basis for contact force field $\bar{\mathbf{P}} = \mathbf{KP}$
while $\mathcal{I} > \varepsilon$ **or** $k < k_{\max}$ **or** $(1 - \mathcal{I}/\mathcal{I}^{\text{old}} < \theta)$ **do**
  for $t_{1 \leqslant k \leqslant m}$ **do**
    Solve local stage with $(\mathbf{v}(t_k), \boldsymbol{\lambda}(t_k))$ as input to get $(\hat{\mathbf{v}}(t_k), \hat{\boldsymbol{\lambda}}(t_k))$ satisfying $\mathcal{R}(\hat{\mathbf{v}}(t_k), \hat{\boldsymbol{\lambda}}(t_k)) = 0$
  end
  for $t_{1 \leqslant k \leqslant m}$ **do**
    Compute $\mathbf{res}_{\text{sd}}(t_k) = \hat{\boldsymbol{\lambda}}(t_k) - \boldsymbol{\lambda}(t_k) + \mathbf{k}(\hat{\mathbf{v}}(t_k) - \mathbf{v}(t_k))$
    Compute time functions by solving $[\mathbf{R}(\mathbf{K} + \mathbf{B}^T\mathbf{kB})\mathbf{P}]\boldsymbol{\psi}^T(t_k) = \mathbf{RB}^T\mathbf{res}_{\text{sd}}(t_k)$
  end
  Update solution $\mathbf{s} = \begin{cases} \mathbf{u} \leftarrow \mathbf{u} + \mathbf{P}\boldsymbol{\psi}^T \\ \boldsymbol{\lambda} \leftarrow \boldsymbol{\lambda} + \bar{\mathbf{P}}\boldsymbol{\psi}^T \end{cases}$
  Save previous convergence criterion $\mathcal{I}^{\text{old}} \leftarrow \mathcal{I}$
  Compute convergence criterion $\mathcal{I}$
  Increment iterations $k \leftarrow k + 1$
end

In practice, the LATIN-ROM computational cost is very low but depends strongly on the quality of the considered reduced basis. Given a basis of size K, the unknowns for the reduced global stage are K time functions sizing m (*i.e.* the number of time steps). The $K \times K$ system has to be solved m times (corresponding to each time step).

According to the above previously LATIN-ROM formulation, the solution is repre-

sented using a space-time separated form with prescribed space functions. To reduce once more the computational charge, one can prescribe not only the space modes but also the time modes. Then, the unknowns of this new reduced global stage would be scalar amplitudes for each time-space mode. In the following, the corresponding formulation is given. Starting from the global stage formulation to search corrective increments:

$$\begin{cases} \mathbf{K}\boldsymbol{\Delta}\check{\mathbf{u}} = \mathbf{B}^T\boldsymbol{\Delta}\check{\boldsymbol{\lambda}} \\ \boldsymbol{\Delta}\check{\mathbf{v}} = \mathbf{B}\boldsymbol{\Delta}\check{\mathbf{u}} \\ \boldsymbol{\Delta}\check{\boldsymbol{\lambda}} + \mathbf{k}\boldsymbol{\Delta}\check{\mathbf{v}} - \mathbf{res}_{\text{sd}} = 0 \end{cases} \tag{4.1}$$

where $\mathbf{res}_{\text{sd}} = \hat{\boldsymbol{\lambda}} - \check{\boldsymbol{\lambda}} + \mathbf{k}(\hat{\mathbf{v}} - \check{\mathbf{v}})$ is known at this stage, one looks for increments written into a separated form with p prescribed space-time modes, say $\mathbf{P}_i\Psi_i(t)$. Space-time modes are chosen to be "orthonormal" as follows:

$$\Big(\mathbf{P}_i\Psi_i(t) \mid \mathbf{P}_j\Psi_j(t)\Big) = \mathbf{P}_j^T\mathbf{P}_i \int_0^T \Psi_i(t)\Psi_j(t)\,\mathrm{d}t = \delta_{ij} \tag{4.2}$$

The space-time separated form for corrective increments reads:

$$\begin{cases} \boldsymbol{\Delta}\check{\mathbf{u}} = \sum_i^p a_i\mathbf{P}_i\Psi_i(t) = \mathbf{P}\mathbf{A}\boldsymbol{\Psi}^T \\ \boldsymbol{\Delta}\check{\boldsymbol{\lambda}} = \sum_i^p a_i\bar{\mathbf{P}}_i\Psi_i(t) = \bar{\mathbf{P}}\mathbf{A}\boldsymbol{\Psi}^T \end{cases} \tag{4.3}$$

where a_i are unknowns (amplitudes) and $\mathbf{A} = \mathrm{diag}(\mathbf{a})$ is a diagonal matrix including each a_i for entries. Note that, to ensure the internal balance (see conditions on equation (2.11)), time functions are chosen equal. Applying a Galerkin weak form modeling method using a trial function $\omega^\star = a_j^\star\mathbf{P}_j\Psi_j(t)$ provides the value a_j:

$$a_j = \Big(\mathbf{P}_j^T \int_0^T \mathbf{B}^T\mathbf{res}_{\text{sd}}\Psi_j(t)\,\mathrm{d}t\Big)\Big/\Big(\mathbf{P}_j^T(\mathbf{K} + \mathbf{B}^T\mathbf{k}\mathbf{B})\mathbf{P}_j\Big) \tag{4.4}$$

With this new formulation prescribing K space-time modes, global stage consists in performing K scalar divisions and hence the computational effort is greatly reduced. The corresponding pseudo code for this reduced global stage is given in algorithm 13. Nonetheless, one has to be advised that such a reduction is very tough and restrictive. Indeed, even if a great computational work can be spared, rebuilding solutions with prescribed space-time mode can be difficult due to the lack in freedom to adjust the accuracy.

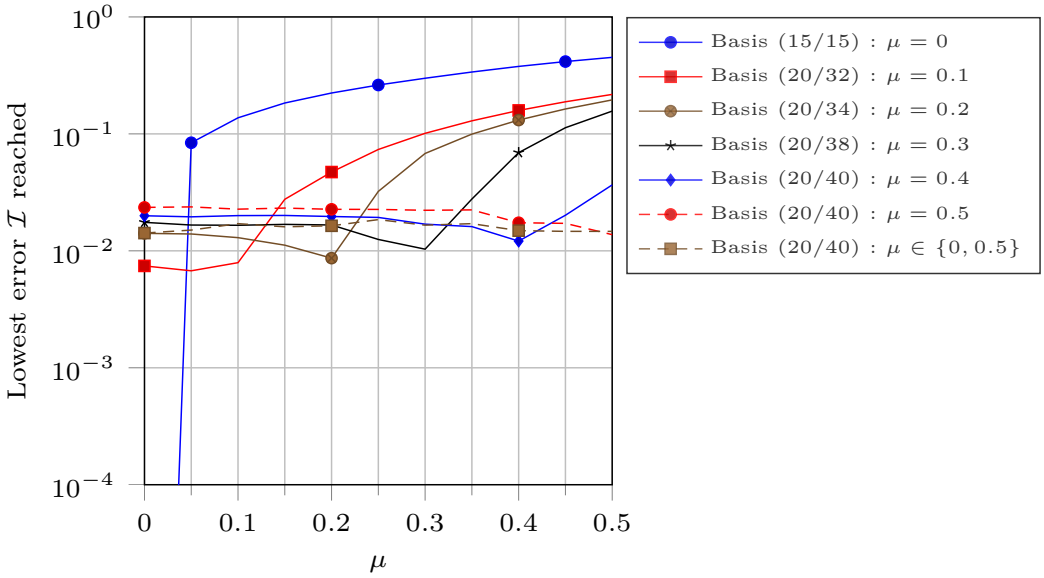
To illustrate both LATIN-ROM formulations with prescribed space modes or with prescribed space-time modes, the extrusion problem is solved for different frictional coefficients $\mu \in I_\mu = [0, 0.5]$. Different choices for the basis are studied: from SVD modes of a single frictional solution or from a combination of SVD modes for different solutions.

Accuracy of solutions obtained with both LATIN-ROM formulations are plot on figures 4.1 and 4.2. Accuracy criterion is based on the LATIN error $\mathcal{I}$ defined on equation (1.54). To remind some order of magnitudes:

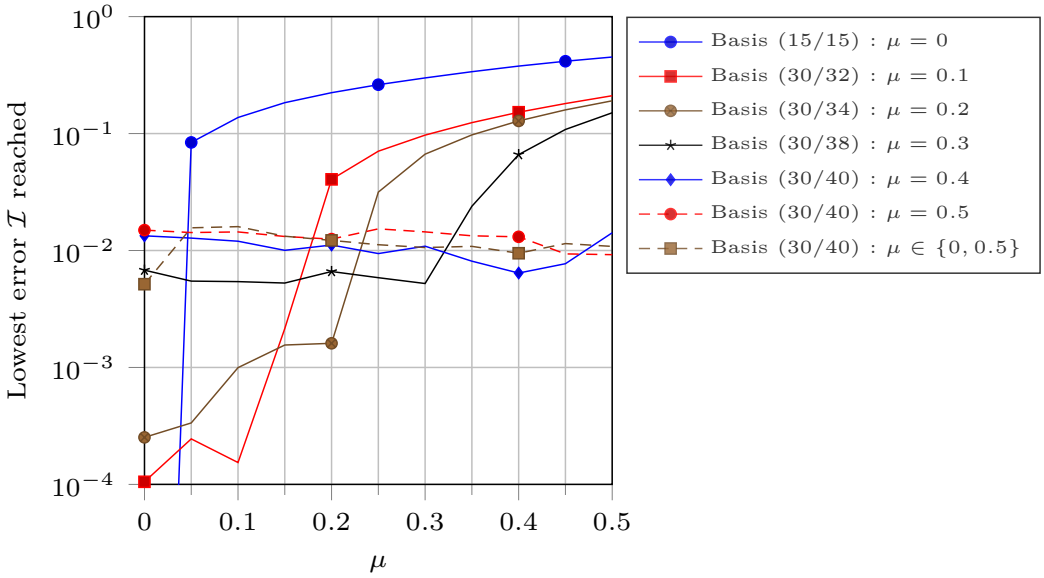
Algorithm 13: The LATIN-ROM method with prescribed space-time modes.

Input:
First guess of the solution $\mathbf{s}_0 = (\mathbf{u}_0, \boldsymbol{\lambda}_0)$
Precision ε
Maximum number of LATIN-ROM iterations $k_{\max}$
Matrix of vector basis $\mathbf{P}$
Stagnation criterion θ

Initialization $k = 0$
Compute basis for contact force field $\bar{\mathbf{P}} = \mathbf{K}\mathbf{P}$
while $\mathcal{I} > \varepsilon$ **or** $k < k_{\max}$ **or** $(1 - \mathcal{I}/\mathcal{I}^{\text{old}} < \theta)$ **do**
  for $t_{1 \leqslant k \leqslant m}$ **do**
    Solve local stage with $\big(\mathbf{v}(t_k), \boldsymbol{\lambda}(t_k)\big)$ as input to get $\big(\hat{\mathbf{v}}(t_k), \hat{\boldsymbol{\lambda}}(t_k)\big)$ satisfying $\mathcal{R}\big(\hat{\mathbf{v}}(t_k), \hat{\boldsymbol{\lambda}}(t_k)\big) = 0$
    Compute $\mathbf{res}_{\text{sd}}(t_k) = \hat{\boldsymbol{\lambda}}(t_k) - \boldsymbol{\lambda}(t_k) + \mathbf{k}\big(\hat{\mathbf{v}}(t_k) - \mathbf{v}(t_k)\big)$
  end
  Compute amplitudes by solving $[\mathbf{P}^T(\mathbf{K} + \mathbf{B}^T\mathbf{k}\mathbf{B})\mathbf{P}]\mathbf{a} = \mathbf{P}^T\mathbf{B}^T\mathbf{res}_{\text{sd}}\boldsymbol{\Psi}$
  Update solution $\mathbf{s} = \begin{cases} \mathbf{u} \leftarrow \mathbf{u} + \mathbf{P}\,\text{diag}(\mathbf{a})\boldsymbol{\Psi}^T \\ \boldsymbol{\lambda} \leftarrow \boldsymbol{\lambda} + \bar{\mathbf{P}}\,\text{diag}(\mathbf{a})\boldsymbol{\Psi}^T \end{cases}$
  Save previous convergence criterion $\mathcal{I}^{\text{old}} \leftarrow \mathcal{I}$
  Compute convergence criterion $\mathcal{I}$
  Increment iterations $k \leftarrow k + 1$
end

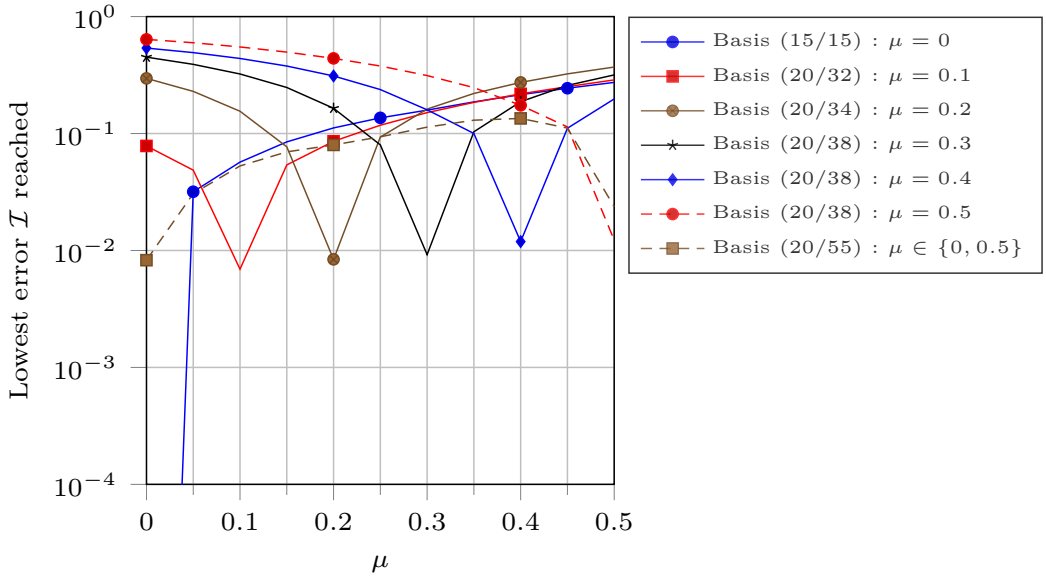


(a) LATIN-ROM with 20 space modes. The frictionless solution counts only 15 SVD modes.

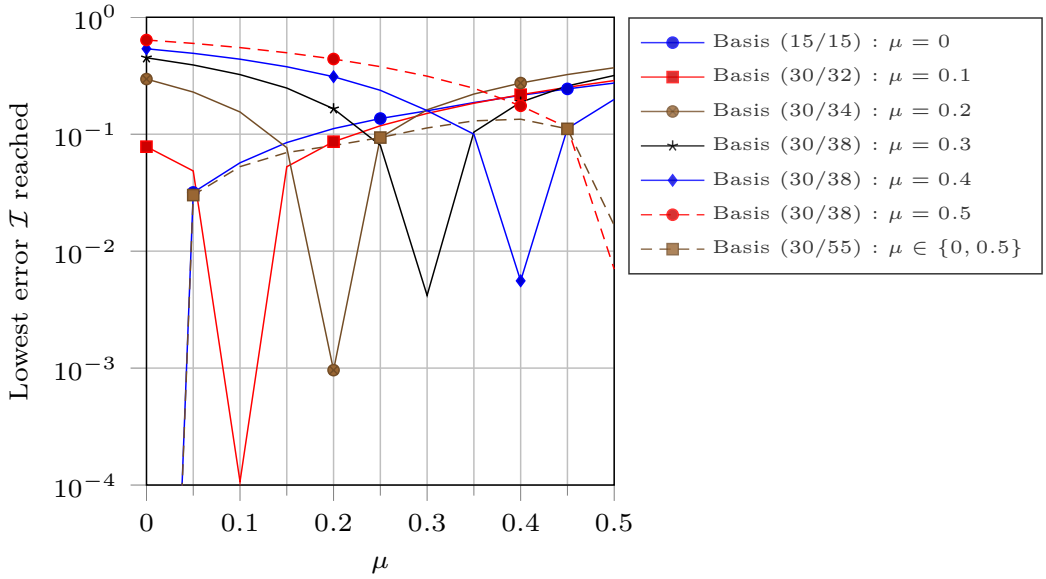


(b) LATIN-ROM with 30 space modes

Figure 4.1 – Lowest error reached by the LATIN-ROM with prescribed space modes to solve the same problem with different frictional coefficients. Considered reduced basis are obtained from different solution. The range for the frictional coefficient $I_\mu = [0, 0.5]$ is sampled uniformly using a step of 0.05. For instance, the second legend entry indicates the error plot with a basis obtained from a frictional solution with $\mu = 0.1$ of including the 20 first SVD modes among the 32 available ones.



(a) LATIN-ROM with 20 space-time modes. The frictionless solution counts only 15 SVD modes.



(b) LATIN-ROM with 30 space-time modes

Figure 4.2 – Lowest error reached by the LATIN-ROM with prescribed space-time modes to solve the same problem with different frictional coefficients. Considered reduced basis are obtained from different solutions.

- $\mathcal{I} \sim 10^{-1}$ corresponds roughly to a 2 norm relative error of 10^{-2} for the displacement field and 10^{-1} for the contact force field

- $\mathcal{I} \sim 10^{-2}$ corresponds roughly to a 2 norm relative error of 10^{-4} for the displacement field and 10^{-2} for the contact force field

In practice, the LATIN-ROM methods require a very few iterations to reach the maximal precision allowed by the considered reduced basis.

As an example, on figure 4.1 case (a) – red curve with square marks – shows the best accuracy reachable with the LATIN-ROM using 20 space modes obtained from the frictional solution $\mu = 0.1$. In this case, the computed frictional solutions for $\mu \leqslant 0.1$ have an error less than 10^{-2}. For $\mu \geq 0.1$, error increases. One can also remark that for $\mu = 0.1$, the recomputed solution error corresponds to the best precision reachable with the 20 optimal modes (see a posteriori analysis in section 1.5.1). Similarly, if one takes the whole modes of a solution, its recalculation will converge to it (see the frictionless case).

The LATIN-ROM with prescribed space modes (figure 4.1) shows interesting performances. Of course, the larger the basis, the better becomes the overall performance but the heavier becomes the computational costs. It seems that the most suited basis is the one obtained from the higher frictional solution ($\mu = 0.5$). This can be explained by the fact that such a solution captures both global trends of whole solutions. On the contrary, the frictionless solution does not capture frictional behavior; therefore, the computations involving its SVD modes are not accurate.

The LATIN-ROM with prescribed space-time modes (figure 4.2) shows also some good performances. As suggested above, this formulation does not improve the accuracy but accelerates and reduces the computational work by prescribing time modes. Furthermore, one can see that this formulation is clearly restrictive and is actually efficient to solve only very short range for μ. For instance, a space-time basis of size 30 obtained from the frictional solution $\mu = 0.4$ will be relevant ($\mathcal{I} < 10^{-1}$) to recompute solutions for $\mu \in [0.35, 0.45]$; elsewhere the error is too large. As this LATIN-ROM formulation requires a very light computational effort, one can afford to consider a larger amount of modes (from different solutions) still keeping in mind that the offline phase becomes more expensive. In this case, a great improvement could be gained: the space-time basis including $15 + 15 = 30$ space-time modes from the solutions $\mu = 0$ and $\mu = 0.5$ is definitely more accurate than the others but remains less accurate than the LATIN-ROM formulation prescribing only space modes.

All in all, both LATIN-ROM formulations show their advantages and their drawbacks but seems to be interesting to tackle frictional problems up to standard level of accuracy. Further developments would consist in obtaining SVD modes through surrogate models, using interpolation methods for snapshots [AF08, ACCF09, AF11, AZF12a], deploying an efficient strategy to select suited design points [MNPP07], or deploying specific methods

such as hyper-reduction to tackle large problems [CFCA13]. To conclude, the challenging issues is the computation of a relevant reduced basis.

4.1.2 The LATIN-PGD extended to parametric problems

The extension of the LATIN-PGD for parametric studies is here discussed. It consists in solving once and for all, a space-time-parameter solution defined on $\Omega \times [0,T] \times I_\mu$. For that purpose, the solution is sought as a low rank tensor approximation, as in equation (3.3). Considering the frictional coefficient as a parameter, the LATIN-PGD space-time-parameter formulation is given in the following. The global stage of the LATIN method yields to solve the following system:

$$\begin{cases} \mathbf{K}\boldsymbol{\Delta u}(t,\mu) = \mathbf{B}^T\boldsymbol{\Delta\lambda}(t,\mu) \\ \boldsymbol{\Delta v}(t,\mu) = \mathbf{B}\boldsymbol{\Delta u}(t,\mu) \\ \boldsymbol{\Delta\lambda}(t,\mu) + \mathbf{k}\boldsymbol{\Delta v}(t,\mu) - \mathbf{res}_{\text{sd}}(t,\mu) = 0 \end{cases} \tag{4.5}$$

with $\mathbf{res}_{\text{sd}}(t,\mu) = \hat{\boldsymbol{\lambda}}(t,\mu) - \boldsymbol{\lambda}(t,\mu) + \mathbf{k}(\hat{\mathbf{v}}(t,\mu) - \mathbf{v}(t,\mu))$ is known at this stage. The low rank tensor approximation for increments reads:

$$\begin{cases} \boldsymbol{\Delta u}(t,\mu) = \mathbf{V}\phi(t)w(\mu) \\ \boldsymbol{\Delta\lambda}(t,\mu) = \mathbf{L}\psi(t)\iota(\mu) \end{cases} \tag{4.6}$$

The internal balance has to be ensured yielding to the following condition for modes:

$$\mathbf{KV}\phi(t)w(\mu) = \mathbf{B}^T\mathbf{L}\psi(t)\iota(\mu) \quad \Rightarrow \quad \forall t, \forall \mu : \begin{cases} \mathbf{KV} = \mathbf{B}^T\mathbf{L} \\ \phi(t) = \psi(t) \\ w(\mu) = \iota(\mu) \end{cases} \tag{4.7}$$

Then, the search direction equation is attempted to be verified at best using a weak form Galerkin method with the trial function $\omega^\star = \mathbf{W}^\star\phi(t)w(\mu) + \mathbf{W}v^\star(t)w(\mu) + \mathbf{W}\phi(t)w^\star(\mu)$ on the domain $\mathcal{D} = [0,T] \times I_\mu$ as follows:

$$\begin{aligned} \forall(\mathbf{W}^\star, v^\star, w^\star) : \int_{\mathcal{D}} \omega^\star\Big(\boldsymbol{\Delta\lambda} + \mathbf{k}\boldsymbol{\Delta v} - \mathbf{res}_{\text{sd}}\Big)\,\mathrm{d}t\,\mathrm{d}\mu = \int_{\mathcal{D}} (\mathbf{W}^\star)^T vw\Big(\mathbf{W}vw - \mathbf{res}_{\text{sd}}\Big)\,\mathrm{d}t\,\mathrm{d}\mu \\ + \int_{\mathcal{D}} v^\star\mathbf{W}^T w\Big(\mathbf{W}vw - \mathbf{res}_{\text{sd}}\Big)\,\mathrm{d}t\,\mathrm{d}\mu + \int_{\mathcal{D}} w^\star\mathbf{W}^T v\Big(\mathbf{W}vw - \mathbf{res}_{\text{sd}}\Big)\,\mathrm{d}t\,\mathrm{d}\mu = 0 \end{aligned} \tag{4.8}$$

with the auxiliary variable $\mathbf{W} = \mathbf{L} + \mathbf{kBV}$. To find suitable modes $\mathbf{W}$, v and w, this non-linear system can be solved using the same method as in section 3.3.4. As three modes has to be solved, three corresponding steps are required:

First step Space vector $\mathbf{W}$ is updated knowing $\phi(t)$ and $w(\mu)$ from previous steps:

$$\mathbf{W} = \Big(\int_{\mathcal{D}} \mathbf{res}_{\text{sd}}\phi(t)w(\mu)\,\mathrm{d}t\,\mathrm{d}\mu\Big) \Big/ \Big(\int_0^T v^2(t)\,\mathrm{d}t \int_{I_\mu} w^2(\mu)\,\mathrm{d}\mu\Big) \tag{4.9}$$

Second step Time vector $\phi(t)$ is updated knowing $\mathbf{W}$ and $w(\mu)$ from previous steps:

$$\phi(t) = (\mathbf{W}^T \int_{I_\mu} \mathbf{res}_{\text{sd}} w(\mu)\,\mathrm{d}\mu) \Big/ \Big(\mathbf{W}^T\mathbf{W} \int_{I_\mu} w^2(\mu)\,\mathrm{d}\mu\Big) \tag{4.10}$$

Third step Parameter vector $w(\mu)$ is updated knowing $\mathbf{W}$ and $\phi(t)$ from previous steps:

$$w(\mu) = (\mathbf{W}^T \int_0^T \mathbf{res}_{\text{sd}} \phi(t)\,\mathrm{d}t) \Big/ \Big(\mathbf{W}^T\mathbf{W} \int_{I_\mu} v^2(t)\,\mathrm{d}t\Big) \tag{4.11}$$

In practice, very few iterations are needed and a normalization is applied for two of these three vectors. Once converged these three vectors are stored and achieve the global stage. The pseudo-code of this method is given in the algorithm 14.

Algorithm 14: LATIN-PGD space-time-parameter enrichment stage.

Input:
$\mathbf{res}_{\text{sd}}(t,\mu)$
Maximum number of iterations $\xi_{\max}$
Output:
New space mode, time mode, parameter mode $\mathbf{W}$, $\phi(t)$, $w(\mu)$

Initialize $\phi(t)$ and $w(\mu)$ (*e.g.* $\phi(t) = w(\mu) = 1$)
for $\xi = 1$ **to** $\xi_{\max}$ **do**
  Compute $\mathbf{W} = \Big(\int_{\mathcal{D}} \mathbf{res}_{\text{sd}} \phi(t) w(\mu)\,\mathrm{d}t\,\mathrm{d}\mu\Big) \Big/ \Big(\int_0^T v^2(t)\,\mathrm{d}t \int_{I_\mu} w^2(\mu)\,\mathrm{d}\mu\Big)$
  Compute $\phi(t) = (\mathbf{W}^T \int_{I_\mu} \mathbf{res}_{\text{sd}} w(\mu)\,\mathrm{d}\mu) \Big/ \Big(\mathbf{W}^T\mathbf{W} \int_{I_\mu} w^2(\mu)\,\mathrm{d}\mu\Big)$
  Compute $w(\mu) = (\mathbf{W}^T \int_0^T \mathbf{res}_{\text{sd}} \phi(t)\,\mathrm{d}t) \Big/ \Big(\mathbf{W}^T\mathbf{W} \int_{I_\mu} v^2(t)\,\mathrm{d}t\Big)$
  Space mode amplitude $\alpha = \|\mathbf{W}\|_2$
  Normalize space mode $\mathbf{W} \leftarrow \mathbf{W}/\alpha$
  Time mode amplitude $\beta = \|\phi(t)\|_2$
  Normalize time mode $\phi(t) \leftarrow \phi(t)/\beta$
  Report amplitude in parameter function $w(\mu) \leftarrow \alpha\beta w(\mu)$
end

These 3 mode vectors are stored once they are found. Space vectors $\mathbf{V}$ and $\mathbf{L}$ can be deduced from stored $\mathbf{W}$ vector using relationships (3.15) in order to rebuild the solution as follows:

$$\mathbf{s} = \begin{cases} \mathbf{u}(t,\mu) = \mathbf{u}_0 + \sum_{i=1}^p \mathbf{V}_i v_i(t) w_i(\mu) \\ \boldsymbol{\lambda}(t,\mu) = \boldsymbol{\lambda}_0 + \sum_{i=1}^p \mathbf{L}_i v_i(t) w_i(\mu) \end{cases} \tag{4.12}$$

Similarly, a preliminary stage can also be formulated by choosing an appropriate trial function (in the same fashion as it was done for the LATIN-PGD space-time formulation) to improve some preexisting modes. To stem the inflation of the PGD expansion and for sake of computational efficiency, a storing strategy has to be planed. Contrary to

the space-time separation formulation, compression and downsizing approach are challenging because tensors at stake are third order ones. Indeed, the optimality for such a PGD expansion is no more related to orthogonality property for modes. Compression and optimal decomposition for high order tensors are challenging. Some decomposition trying to generalize the well known SVD to high order tensors can be found in the literature: HOSVD, PARAFAC, CanDecomp but remains nowadays under development [Bro97, Kol01, KB09, DLDMV00b, DLDMV00a, LDH11].

In the following, each space-time-parameter is stored in a raw once computed. No orthogonal property stands and no downsizing strategy is implemented. The pseudo code of this extended version of the LATIN-PGD is described in algorithm 15.

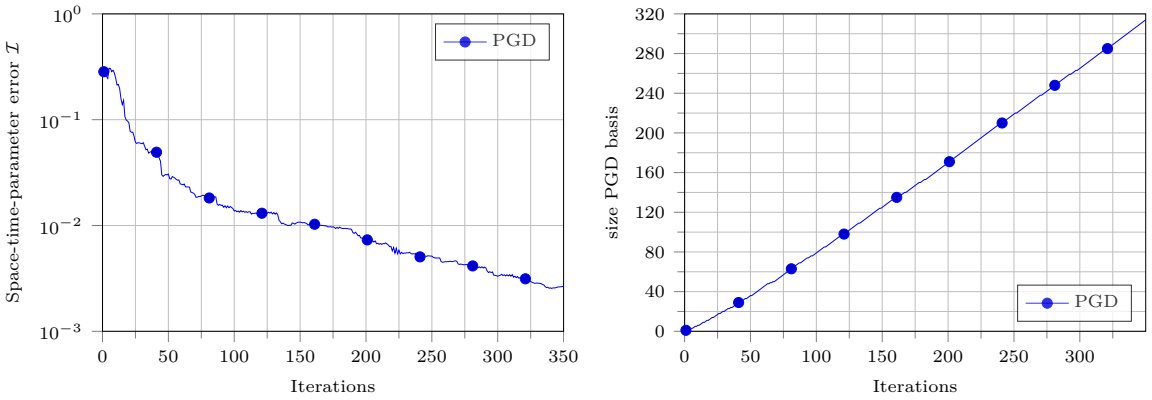
Algorithm 15: The extended LATIN-PGD method for parametric studies.

Data:
Precisions ε, $\varepsilon_{\max}$
Enriching criterion $0 \leqslant \theta \leqslant 1$

Solve the linear elastic solution $\mathbf{K}\mathbf{u}_0 = \mathbf{f}_{\text{ext}}$ and $\boldsymbol{\lambda}_0 = \mathbf{0}$
Initialization of the PGD basis $p = 0$, $(\mathbf{W}_k)_1^p = \emptyset$, $(\phi_k)_1^p = \emptyset$, $(w_k)_1^p = \emptyset$
while $\mathcal{I} > \varepsilon$ **do**
    `** Local stage`
    for $\forall \mu \in I_\mu$ **do**
        for $t_{1 \leqslant k \leqslant m}$ **do**
            Solve the local stage such that $\mathcal{R}(\hat{\mathbf{v}}, \hat{\boldsymbol{\lambda}}) = 0 \rightarrow (\hat{\mathbf{v}}, \hat{\boldsymbol{\lambda}})$
        end
    end
    `** Extended preliminary stage`
    if $p \geqslant 1$ **then**
        Compute $\mathbf{res}_{\text{sd}}$
        for $i = 1$ ***to*** p **do**
            Space modes update $\rightarrow (\mathbf{W}_k)_1^p$
            Update $\mathbf{res}_{\text{sd}}$
            Time modes update $\rightarrow (\phi_k)_1^p$
            Update $\mathbf{res}_{\text{sd}}$
            Parameter modes update $\rightarrow (w_k)_1^p$
            Update $\mathbf{res}_{\text{sd}}$
        end
    end
    `** Enrichment stage`
    if $(1 - \mathcal{I}/\mathcal{I}^{\text{old}} < \theta)$ **then**
        Compute $\mathbf{res}_{\text{sd}}$
        Enrichment stage $\rightarrow \{\mathbf{W}, \phi, w\}$
        Raw enrichment $\rightarrow \{(\mathbf{W}_k)_1^{p+1}, (\phi_k)_1^{p+1}, (w_k)_1^{p+1}\}$
        Increment basis size $p \leftarrow p + 1$
    end
    `** Convergence check`
    Save previous convergence criterion $\mathcal{I}^{\text{old}} \leftarrow \mathcal{I}$
    Compute convergence criterion $\mathcal{I}$
end

This method is applied on the extrusion problem considering $I_\mu = [0, 0.5]$. The convergence criterion defined in (1.54) is extended to take into account the parameter dimension:

$$\mathcal{I} = \max\left(\sqrt{\frac{\|\mathbf{s}_N - \hat{\mathbf{s}}_N\|_\infty^2}{\|\mathbf{s}_N\|_\infty^2 + \|\hat{\mathbf{s}}_N\|_\infty^2}}, \sqrt{\frac{\|\mathbf{s}_T - \hat{\mathbf{s}}_T\|_\infty^2}{\|\mathbf{s}_T\|_\infty^2 + \|\hat{\mathbf{s}}_T\|_\infty^2}}\right)$$
$$\text{with} \quad \|\mathbf{s}_{N,T}\|_\infty^2 = \max_{\partial_3\Omega, t, \mu}\left[\frac{1}{k_{N,T}}\lambda_{N,T}^2 + k_{N,T}u_{N,T}^2\right] \tag{4.13}$$



(a) Evolution of the error

(b) Basis size

Figure 4.3 – Behavior of the extended LATIN-PGD accounting the parameter μ.

The evolution of the error (denoted with $\mathcal{I}$ and relatively to reference solutions obtained by performing a full dimensional LATIN method with appropriate μ) and the evolution of the PGD basis are plotted on figure 4.3.

The extended LATIN-PGD shows promising performances. Indeed, a satisfying level of accuracy, say 10^{-2}, with a very tough convergence criterion is reached. For the tested case, this method is able to compute a priori and at once a solution which would involve 11 separate computations. Moreover, the solution is provided as a low rank tensor approximation which is easy to store and ready to use for post-processing devices.

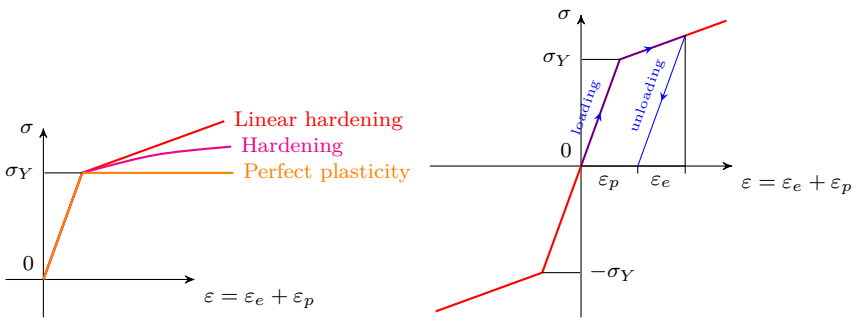
4.2 Extension to material plasticity

The LATIN method is a general solver for non-linear and time dependent problems. Hereinafter, in addition to frictional contact, non-linearity induced by plasticity is accounted. In the following, a LATIN formulation is briefly given and some development paths for a LATIN-PGD formulation are given.

4.2.1 Hardening material and associated flow rule

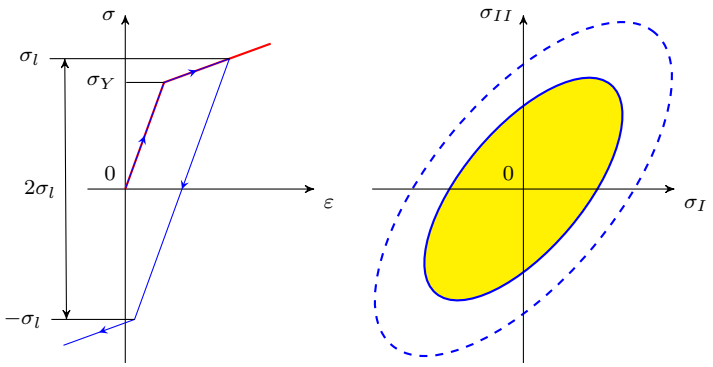
In this section, a brief review of some general elastoplastic constitutive models is given. First, we assume that the plastic regime is governed by a hardening law (see figure 4.4). If an elastoplastic body is taken beyond the yield stress denoted by σ_Y up to a given stress $\sigma_l > \sigma_Y$, it begins to derform plastically. When the load is released, a permanent plastic deformation, say ε_p, remains. If the body is loaded again, no additional plastic deformation occurs until the stress reaches σ_l: the material is hardened.

In the biaxial stress plane (see figure 4.5), any stress state in the initial yield surface (yellow) is in the elastic region. Once the body is taken beyond the initial yield surface,



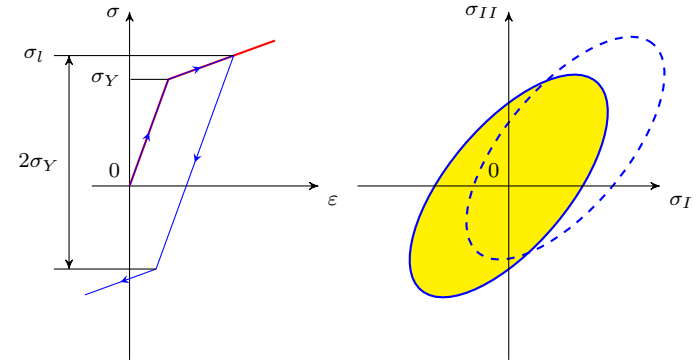
(a) Hardening laws (b) Uniaxial stress-strain law

Figure 4.4 – Pure tensile test : uniaxial stress-strain curves.



(a) Uniaxial stress-strain law (isotropic hardening)

(b) Evolution of the yield surface in the biaxial stress plane (isotropic hardening)



(c) Uniaxial stress-strain law (kinematic hardening)

(d) Evolution of the yield surface in the biaxial stress plane (kinematic hardening)

Figure 4.5 – Isotropic vs. kinematic hardening rule.

it experiences plastic deformation.

- With an isotropic hardening rule, the center of the yield surface remains the same but it expands. Any stress state inside the new yield surface will experience elastic deformation. New plastic deformations occurs when the stress state reaches the new surface (delimited by the dashed region).

- With a kinematic hardening rule, the center of the yield surface translates but its shape remains the same. Any stress state inside the new yield surface will experience elastic deformation. New plastic deformation occurs when the stress state reaches the new surface (delimited by the dashed region). This rule is able to model the Bauschinger effect and similar response where a hardening in tension will lead to a softening in a subsequent compression.

- A more general hardening consists in combining both isotropic and kinematic rules.

In metal plasticity, the plastic strain increment is usually assumed to have the same principal directions as the deviatoric stress tensor and it is normal to the yield surface (normality condition). For rock and soil plasticity theory, such an assumption is not suited (overestimation of the plastic volume expansion [Man66]). In the following, a linear isotropic hardening rule is selected. Then, the yield surface is defined with the following plastic potential:

$$f(\boldsymbol{\sigma}, R) = J(\boldsymbol{\sigma}) - [\sigma_Y + R(p)] \quad \begin{cases} f(\boldsymbol{\sigma}, R) \leqslant 0 & \text{elastic regime} \\ f(\boldsymbol{\sigma}, R) = 0 & \text{at yield} \end{cases} \tag{4.14}$$

with J a yield criterion (taken equal to the norm of the deviatoric stress), $R(p)$ a function governing the hardening law and p the cumulative plastic strain. As a linear hardening law is considered, the function $R(p) = hp$ is linear where h is the hardening modulus. The normality condition can be expressed with the following flow rule:

$$\begin{cases} \mathrm{d}\boldsymbol{\varepsilon}_p = \lambda \frac{\partial f}{\partial \boldsymbol{\sigma}} \\ \mathrm{d}p = -\lambda \frac{\partial f}{\partial R} \end{cases} \quad \text{and} \quad \begin{cases} \text{if } f(\boldsymbol{\sigma}, R) < 0 \text{ then } \lambda = 0 \\ \text{if } f(\boldsymbol{\sigma}, R) = 0 \text{ then } \lambda \geqslant 0 \end{cases} \quad \Leftrightarrow \quad \begin{cases} f(\boldsymbol{\sigma}, R) \leqslant 0 \\ \lambda \geqslant 0 \\ \lambda f(\boldsymbol{\sigma}, R) = 0 \end{cases} \tag{4.15}$$

with λ the plastic multiplier. This law is an associated (or standard) flow rule [Hil58, Dru57, SH] and has convenient mathematical properties related to the convex analysis. Nevertheless, such a law is not universal and other classes of flow rules (*e.g.* non-associated flow rules) were suggested. In the following, this associated flow rule is considered.

In addition, in order to eliminate the unknown λ, one can consider the Prager consis-

tency condition [Pra49]:

$$\begin{cases} f(\boldsymbol{\sigma},R)<0 \text{ then } \lambda=0 \\ f(\boldsymbol{\sigma},R)=0 \text{ then } \mathrm{d}f=0 \end{cases} \Leftrightarrow \begin{cases} \text{If } f(\boldsymbol{\sigma},R)<0 \text{ then } \lambda=0 \\ \text{If } f(\boldsymbol{\sigma},R)=0 \text{ then } \begin{cases} \mathrm{d}f \geqslant 0 \\ \lambda \geqslant 0 \\ \lambda\,\mathrm{d}f=0 \end{cases} \end{cases} \tag{4.16}$$

with $\mathrm{d}f = \frac{\partial f}{\partial \boldsymbol{\sigma}} : \mathrm{d}\boldsymbol{\sigma} + \frac{\partial f}{\partial R} : \mathrm{d}R$. As mentioned in the first chapter, those conditions can be obtained from (4.15) using the Moreau's viability lemma [Mor88].

4.2.2 A LATIN formulation for frictional contact with elastoplasticity

In this section a LATIN formulation (without space-time representation) accounting both frictional contact and the above plastic constitutive law is proposed. It still consists in a two stage iterative and non-incremental process dealing separately with the linear (internal balance) and the non-linear behavior (both frictional contact and plasticity). For a non-linear problem involving only plasticity, several formulations are given in [Lad99]. Each stage aims at solving for fields defined on the space-time domain.

Global stage (linear stage)

Given the previous solution of the local stage denoted by $\hat{\mathbf{s}} = (\hat{\boldsymbol{\sigma}}, \hat{\boldsymbol{\varepsilon}}, \hat{\mathbf{v}}, \hat{\boldsymbol{\lambda}})$, a solution $\mathbf{s} = (\boldsymbol{\sigma}, \boldsymbol{\varepsilon}, \mathbf{v}, \boldsymbol{\lambda})$ verifying internal balance, static admissibility, kinematic admissibility is sought for. A correction formulation leads to solve the following system:

$$\begin{cases} \mathbf{B}_\sigma^T \boldsymbol{\sigma} = \mathbf{f}_{\text{ext}} + \mathbf{B}^T \boldsymbol{\lambda} \\ \boldsymbol{\varepsilon} = \mathbf{B}_\varepsilon \mathbf{u} \\ \mathbf{v} = \mathbf{B}\mathbf{u} \end{cases} \quad \text{and} \quad \begin{cases} \boldsymbol{\sigma} = \boldsymbol{\sigma}_{i-1} + \Delta\boldsymbol{\sigma} \\ \mathbf{u} = \mathbf{u}_{i-1} + \Delta\mathbf{u} \\ \boldsymbol{\lambda} = \boldsymbol{\lambda}_{i-1} + \Delta\boldsymbol{\lambda} \end{cases} \tag{4.17}$$

with $\mathbf{B}_\varepsilon$ the gradient operator, $\mathbf{B}_\sigma$ the stress equilibrium operator. Moreover, the following search direction equations are introduced:

$$\begin{cases} \boldsymbol{\sigma} - \mathbf{D}\boldsymbol{\varepsilon} = \hat{\boldsymbol{\sigma}} - \mathbf{D}\hat{\boldsymbol{\varepsilon}} \\ \boldsymbol{\lambda} + \mathbf{k}\mathbf{v} = \hat{\boldsymbol{\lambda}} + \mathbf{k}\hat{\mathbf{v}} \end{cases} \tag{4.18}$$

where the search direction parameter for strain-stress pair is taken equal to the Hooke's law operator $\mathbf{D}$. All in all, the system providing the suited correction $\Delta\mathbf{s} = (\Delta\boldsymbol{\sigma}, \Delta\boldsymbol{\varepsilon}, \Delta\mathbf{v}, \Delta\boldsymbol{\lambda})$ is:

$$\begin{cases} \mathbf{B}_\sigma^T \Delta\boldsymbol{\sigma} = \mathbf{B}^T \Delta\boldsymbol{\lambda} \\ \Delta\boldsymbol{\sigma} - \mathbf{D}\mathbf{B}_\varepsilon \Delta\mathbf{u} = \mathbf{res}_P \\ \Delta\boldsymbol{\lambda} + \mathbf{k}\mathbf{B}\Delta\mathbf{u} = \mathbf{res}_C \\ \Delta\mathbf{v} = \mathbf{B}\Delta\mathbf{u} \\ \Delta\boldsymbol{\varepsilon} = \mathbf{B}_\varepsilon \Delta\mathbf{u} \end{cases} \Leftrightarrow \begin{cases} [\mathbf{K} + \mathbf{B}^T\mathbf{k}\mathbf{B}]\Delta\mathbf{u} = \mathbf{B}^T\mathbf{res}_C - \mathbf{B}_\sigma^T\mathbf{res}_P \\ \Delta\mathbf{v} = \mathbf{B}\Delta\mathbf{u} \\ \Delta\boldsymbol{\varepsilon} = \mathbf{B}_\varepsilon\Delta\mathbf{u} \\ \Delta\boldsymbol{\sigma} = \mathbf{res}_P + \mathbf{D}\mathbf{B}_\varepsilon\Delta\mathbf{u} \end{cases} \tag{4.19}$$

with $\mathbf{res}_P = (\hat{\boldsymbol{\sigma}} - \mathbf{D}\hat{\boldsymbol{\varepsilon}}) - (\boldsymbol{\sigma}_{i-1} - \mathbf{D}\mathbf{B}_\varepsilon \mathbf{u}_{i-1})$, $\mathbf{res}_C = (\hat{\boldsymbol{\lambda}} + \mathbf{k}\hat{\mathbf{v}}) - (\boldsymbol{\lambda}_{i-1} + \mathbf{k}\mathbf{B}\mathbf{u}_{i-1})$ and the stiffness operator $\mathbf{K} = \mathbf{B}_\sigma \mathbf{D}\mathbf{B}_\varepsilon$. Once the global stage is solved, the solution $\mathbf{s} = \mathbf{s}_{i-1} + \Delta\mathbf{s}$ is deduced.

Non-linear stage (local stage)

Given $\mathbf{s} = (\boldsymbol{\sigma}, \boldsymbol{\varepsilon}, \mathbf{v}, \boldsymbol{\lambda})$, the non-linear stage aims to compute $\hat{\mathbf{s}} = (\hat{\boldsymbol{\sigma}}, \hat{\boldsymbol{\varepsilon}}, \hat{\mathbf{v}}, \hat{\boldsymbol{\lambda}})$ verifying frictional contact conditions and plasticity behavior. Frictional conditions and plasticity behavior are tackled separately and independently. Therefore, the local stage defined for frictional contact in 1.6.2 is used to find $(\hat{\mathbf{v}}, \hat{\boldsymbol{\lambda}})$. Hereinafter, the problem to find $(\hat{\boldsymbol{\sigma}}, \hat{\boldsymbol{\varepsilon}})$ is formulated.

First, the associated plasticity flow rule accounting an isotropic linear hardening rule yields to:

$$\begin{cases} \mathrm{d}\hat{\boldsymbol{\varepsilon}}_p = \hat{\lambda}\frac{\partial f}{\partial \hat{\boldsymbol{\sigma}}} \\ \mathrm{d}\hat{p} = -\hat{\lambda}\frac{\partial f}{\partial \hat{R}} \end{cases} \quad \text{with} \quad \begin{cases} f(\hat{\boldsymbol{\sigma}}, \hat{R}) \leqslant 0 \\ \hat{\lambda} \geqslant 0 \\ \hat{\lambda} f(\hat{\boldsymbol{\sigma}}, \hat{R}) = 0 \end{cases} \tag{4.20}$$

with the plastic potential defined as follows:

$$f(\hat{\boldsymbol{\sigma}}, \hat{R}) = \| \mathbf{dev}(\hat{\boldsymbol{\sigma}}) \| - (\sigma_Y + \hat{R}) \tag{4.21}$$

with $\mathbf{dev}(\hat{\boldsymbol{\sigma}})$ is the deviatoric part of the stress tensor $\hat{\boldsymbol{\sigma}}$ and $\hat{R} = h\hat{p}$. The deviatoric norm is such that $\| \mathbf{dev}(\hat{\boldsymbol{\sigma}}) \| = \sqrt{\frac{3}{2}\,\mathrm{trace}[\mathbf{dev}(\hat{\boldsymbol{\sigma}})\,\mathbf{dev}(\hat{\boldsymbol{\sigma}})]}$. Furthermore, the Hooke's law states:

$$\hat{\boldsymbol{\sigma}} = \mathbf{D}(\hat{\boldsymbol{\varepsilon}} - \hat{\boldsymbol{\varepsilon}}_p) \tag{4.22}$$

And finally, the search direction equation is introduced:

$$\hat{\boldsymbol{\sigma}} + \mathbf{D}\hat{\boldsymbol{\varepsilon}} = \boldsymbol{\sigma} + \mathbf{D}\boldsymbol{\varepsilon} \;\Leftrightarrow\; \hat{\boldsymbol{\sigma}} = \boldsymbol{\sigma} + \mathbf{D}\boldsymbol{\varepsilon} - \mathbf{D}\hat{\boldsymbol{\varepsilon}} \tag{4.23}$$

Combining the Hooke's law and the search direction equation yields to:

$$\hat{\boldsymbol{\sigma}} = \tfrac{1}{2}[(\boldsymbol{\sigma} + \mathbf{D}\boldsymbol{\varepsilon}) - \mathbf{D}\hat{\boldsymbol{\varepsilon}}_p] \tag{4.24}$$

Then, the θ-method scheme [SP91, BGH00] is introduced to express $\hat{\boldsymbol{\varepsilon}}_p$ (at time step t_k) with anterior quantities (at the previous time steps t_{k-1}) denoted with superscript $k-1$:

$$\hat{\boldsymbol{\sigma}} = \tfrac{1}{2}[(\boldsymbol{\sigma} + \mathbf{D}\boldsymbol{\varepsilon}) - \mathbf{D}(\hat{\mathbf{e}}_p + \theta\Delta t\, \mathrm{d}\hat{\boldsymbol{\varepsilon}}_p)] = \tfrac{1}{2}\Big[(\boldsymbol{\sigma} + \mathbf{D}\boldsymbol{\varepsilon}) - \mathbf{D}\hat{\mathbf{e}}_p - \theta\Delta t\hat{\lambda}\mathbf{D}\frac{\partial f}{\partial \hat{\boldsymbol{\sigma}}}\Big] \tag{4.25}$$

with $\hat{\mathbf{e}}_p = \hat{\boldsymbol{\varepsilon}}_p^{(k-1)} + (1-\theta)\Delta t\, \mathrm{d}\hat{\boldsymbol{\varepsilon}}_p^{(k-1)}$ and $\frac{\partial \hat{f}}{\partial \hat{\boldsymbol{\sigma}}} = \frac{\mathbf{dev}(\hat{\boldsymbol{\sigma}})}{\| \mathbf{dev}(\hat{\boldsymbol{\sigma}}) \|}$. Then, the deviatoric part of $\hat{\boldsymbol{\sigma}}$ can be deduced denoting by G the shear modulus:

$$\begin{aligned} \mathbf{dev}(\hat{\boldsymbol{\sigma}}) &= \mathbf{dev}\left(\tfrac{1}{2}\Big[(\boldsymbol{\sigma} + \mathbf{D}\boldsymbol{\varepsilon}) - \mathbf{D}\hat{\mathbf{e}}_p - \theta\Delta t\hat{\lambda}\mathbf{D}\frac{\hat{\boldsymbol{\sigma}}^D}{\|\hat{\boldsymbol{\sigma}}^D\|}\Big]\right) \\ &= \mathbf{dev}\left(\tfrac{1}{2}\Big[(\boldsymbol{\sigma} + \mathbf{D}\boldsymbol{\varepsilon}) - \mathbf{D}\hat{\mathbf{e}}_p - 2\theta\Delta t\hat{\lambda}G\frac{\hat{\boldsymbol{\sigma}}^D}{\|\hat{\boldsymbol{\sigma}}^D\|}\Big]\right) \end{aligned} \tag{4.26}$$

One can compute the norm the above deviatoric part yielding to:

$$\|\,\mathbf{dev}(\hat{\boldsymbol{\sigma}})\| = \|\tfrac{1}{2}\,\mathbf{dev}(\boldsymbol{\sigma} + \mathbf{D}\boldsymbol{\varepsilon} - \mathbf{D}\hat{\mathbf{e}}_p)\| - \theta\Delta t\hat{\lambda}G \tag{4.27}$$

In the same fashion, the hardening laws can be rewritten with the θ-method scheme:

$$\hat{R} = h\hat{p} = h(\hat{\pi} + \theta\Delta t\,\mathrm{d}\hat{p}) = h\hat{\pi} - h\hat{\lambda}\theta\Delta t\frac{\partial f}{\partial \hat{R}} \tag{4.28}$$

with $\hat{\pi} = \hat{p}^{(k-1)} + (1-\theta)\Delta t\,\mathrm{d}\hat{p}^{(k-1)}$ and $\frac{\partial f}{\partial \hat{R}} = -1$. The plastic potential is then deduced and reads:

$$\hat{f} = \|\,\mathbf{dev}(\hat{\boldsymbol{\sigma}})\| - (\sigma_Y + \hat{R}) = \gamma - \hat{\lambda}(\theta\Delta tG + h\theta\Delta t) \tag{4.29}$$

with $\gamma = \|\frac{1}{2}\,\mathbf{dev}(\boldsymbol{\sigma} + \mathbf{D}\boldsymbol{\varepsilon} - \mathbf{D}\hat{\mathbf{e}}_p)\| - h\hat{\pi} - \sigma_Y = \hat{f} + \hat{\lambda}(\theta\Delta tG + h\theta\Delta t)$. From γ, the plastic multiplier is deduced using complementarity condition:

$$\begin{cases} \text{If } \gamma < 0 \Rightarrow \hat{f} < 0 \Rightarrow \hat{\lambda} = 0 \\ \text{If } \gamma > 0 \Rightarrow \hat{f} = 0 \Rightarrow \hat{\lambda} = \frac{\gamma}{\theta\Delta tG + h\theta\Delta t} \end{cases} \tag{4.30}$$

Once $\hat{\lambda}$ found, $\hat{\mathbf{s}}$ is easily deduced from the above relationship.

Initialization and stopping criterion

An error indicator for the plastic behavior was proposed in [RNB13] and reads:

$$\mathcal{I}_P = \max\left(\frac{\|\hat{\boldsymbol{\varepsilon}} - \boldsymbol{\varepsilon}\|}{\|\hat{\boldsymbol{\varepsilon}} + \boldsymbol{\varepsilon}\|}, \frac{\|\hat{\boldsymbol{\sigma}} - \boldsymbol{\sigma}\|}{\|\hat{\boldsymbol{\sigma}} + \boldsymbol{\sigma}\|}\right) \tag{4.31}$$

Then, the error indicator for the formulated LATIN method for elastoplastic body with frictional contact is defined as follows:

$$\mathcal{I}_{C+P} = \max(\mathcal{I}, \mathcal{I}_P) \tag{4.32}$$

To initialize the LATIN method, $\mathbf{s}_0 = (\boldsymbol{\sigma}_0, \boldsymbol{\varepsilon}_0, \mathbf{v}_0, \boldsymbol{\lambda}_0)$ is computed as follows:

$$\mathbf{K}\mathbf{u}_0 = \mathbf{f}_{\text{ext}} \quad \text{and} \quad \{\mathbf{v}_0 = \mathbf{B}\mathbf{u}_0, \boldsymbol{\varepsilon}_0 = \mathbf{B}_\varepsilon\mathbf{u}_0, \boldsymbol{\sigma}_0 = \mathbf{D}\boldsymbol{\varepsilon}_0, \boldsymbol{\lambda}_0 = \mathbf{0}\} \tag{4.33}$$

with $\mathbf{u}_0$ is kinematically admissible.

Pseudo-code

To sum up the whole suggested formulation, the pseudo-code of this LATIN formulation is given on algorithm 16.

Algorithm 16: The LATIN method for elastoplastic frictional contact problem.

Data:
Precision ε
Maximum number of iterations $i_{\max}$
Time scheme parameter θ (*e.g.* $\theta = \frac{1}{2}$)

Solve the linear elastic solution $\mathbf{s}_0$
Initialization $i = 0$
while $\mathcal{I}_{C+P} > \varepsilon$ **do**
  `** Local stage (contact)`
  for $k = 2$ **to** m (time steps) **do**
    Solve the local stage such that $\mathcal{R}(\hat{\mathbf{v}}, \hat{\boldsymbol{\lambda}}) = 0$
  end
  `** Local stage (plasticity)`
  Set initial condition for $(\hat{\boldsymbol{\sigma}}^{(1)}, \hat{\boldsymbol{\varepsilon}}^{(1)}) = (\hat{\boldsymbol{\sigma}}_0^{(1)}, \hat{\boldsymbol{\varepsilon}}_0^{(1)})$ and $p_0^{(1)} = 0$ and $\hat{\boldsymbol{\varepsilon}}_p^{(1)} = \mathbf{0}$
  Initialize $\hat{\mathbf{e}}_p = \mathbf{0}$ and $\hat{\pi} = 0$
  for $k = 2$ **to** m (time steps) **do**
    Compute γ and deduce $\hat{\lambda}$ using (4.30)
    Compute $\mathbf{dev}(\hat{\boldsymbol{\sigma}})$ using (4.26)
    Compute $\mathrm{d}\hat{\boldsymbol{\varepsilon}}_p^{(k)} = \hat{\lambda} \frac{\mathbf{dev}(\hat{\boldsymbol{\sigma}}^{(k)})}{\|\mathbf{dev}(\hat{\boldsymbol{\sigma}}^{(k)})\|}$
    Compute $\hat{\boldsymbol{\varepsilon}}_p^{(k)} = \hat{\mathbf{e}}_p + \theta \Delta t \, \mathrm{d}\hat{\boldsymbol{\varepsilon}}_p^{(k)}$
    Compute $\hat{\boldsymbol{\sigma}}^{(k)} = \frac{1}{2}[(\boldsymbol{\sigma}^{(k)} + \mathbf{D}\boldsymbol{\varepsilon}^{(k)}) - \mathbf{D}\hat{\boldsymbol{\varepsilon}}_p^{(k)}]$
    Compute $\hat{\boldsymbol{\varepsilon}}^{(k)} = \mathbf{D}^{-1}[\boldsymbol{\sigma}^{(k)} + \mathbf{D}\boldsymbol{\varepsilon}^{(k)} - \hat{\boldsymbol{\sigma}}^{(k)}]$
    Compute $\hat{p}^{(k)} = \hat{\pi} + \hat{\lambda}\theta\Delta t$
    Increment $\hat{\mathbf{e}}_p \leftarrow \hat{\mathbf{e}}_p + \theta\Delta t \, \mathrm{d}\hat{\boldsymbol{\varepsilon}}_p^{(k)}$ and $\hat{\pi} \leftarrow \hat{\pi} + \hat{\lambda}\theta\Delta t$
  end
  `** Global stage`
  for $k = 2$ **to** m (time steps) **do**
    Compute $\mathbf{res}_P^{(k)} = (\hat{\boldsymbol{\sigma}}^{(k)} - \mathbf{D}\hat{\boldsymbol{\varepsilon}}^{(k)}) - (\boldsymbol{\sigma}_{i-1}^{(k)} - \mathbf{D}\mathbf{B}_\varepsilon \mathbf{u}_{i-1}^{(k)})$
    Compute $\mathbf{res}_C^{(k)} = (\hat{\boldsymbol{\lambda}}^{(k)} + \mathbf{k}\hat{\mathbf{v}}^{(k)}) - (\boldsymbol{\lambda}_{i-1}^{(k)} + \mathbf{k}\mathbf{B}\mathbf{u}_{i-1}^{(k)})$
    Solve $[\mathbf{K} + \mathbf{B}^T\mathbf{k}\mathbf{B}]\Delta\mathbf{u}^{(k)} = \mathbf{B}^T\mathbf{res}_C^{(k)} - \mathbf{B}_\sigma^T\mathbf{res}_P^{(k)}$
    Compute $\Delta\boldsymbol{\varepsilon}^{(k)} = \mathbf{B}_\varepsilon\Delta\mathbf{u}^{(k)}$
    Compute $\Delta\boldsymbol{\sigma}^{(k)} = \mathbf{res}_P^{(k)} + \mathbf{D}\mathbf{B}_\varepsilon\Delta\mathbf{u}^{(k)}$
    Compute the solution $\mathbf{s}_i^{(k)} = (\boldsymbol{\sigma}_{i-1}^{(k)}, \boldsymbol{\varepsilon}_{i-1}^{(k)}, \mathbf{v}_{i-1}^{(k)}, \boldsymbol{\lambda}_{i-1}^{(k)}) + (\Delta\boldsymbol{\sigma}^{(k)}, \Delta\boldsymbol{\varepsilon}^{(k)}, \Delta\mathbf{v}^{(k)}, \Delta\boldsymbol{\lambda}^{(k)})$
  end
  `** Convergence check`
  Compute convergence criterion $\mathcal{I}_{C+P}$
  Increment $i \leftarrow i + 1$
end

Toward a novel quasi-optimal reduced basis strategy formulation

The global stage requires to solve the expensive linear system defined on (4.19). Moreover, additional variables such as the stress field and the strain field have to be stored arising some memory storage issues. The PGD seems to be again a convenient method to address these issues. Using the same approach as in the previous chapter, the global stage accounting frictional contact and elastoplastic material could be reformulated using the following representation:

$$\begin{cases} \Delta\boldsymbol{\sigma} = \mathbf{S}s(t) \\ \Delta\boldsymbol{\lambda} = \mathbf{L}l(t) \\ \Delta\mathbf{u} = \mathbf{V}v(t) \end{cases} \tag{4.34}$$

Note that the strain field increment is straightforwardly deduced using $\Delta\boldsymbol{\varepsilon} = \mathbf{B}_\varepsilon\mathbf{V}v(t)$. Hereinafter, an enrichment stage is formulated. The internal balance and the above representation has to be ensured leading to following conditions for some modes:

$$\begin{cases} \mathbf{B}_\sigma^T\mathbf{S} = \mathbf{B}^T\mathbf{L} \\ s(t) = l(t) \end{cases} \tag{4.35}$$

search direction equations will be therefore approximated. The challenge of this new formulation is to get a suited strategy to find the most relevant modes ensuring internal balance and approximating at best search direction equation. To address it, the following formulation (a primer) for modes is suggested. First, displacement modes are sought by solving:

$$\{\mathbf{V}, v(t)\} = \arg\min_{\mathbf{V}^\star, v^\star} \left([\mathbf{K} + \mathbf{B}^T\mathbf{k}\mathbf{B}]\mathbf{V}^\star v^\star(t) - (\mathbf{B}^T\mathrm{res}_C - \mathbf{B}_\sigma^T\mathrm{res}_P)\right) \tag{4.36}$$

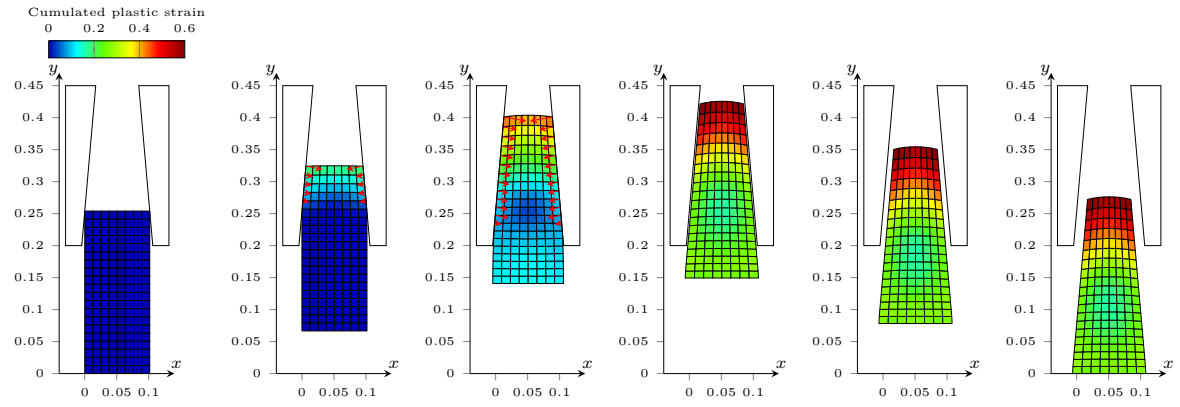
Then, representations for the remaining modes are prescribed as follows:

$$\begin{cases} \Delta\boldsymbol{\sigma} = \mathbf{S}s(t) \\ \Delta\boldsymbol{\lambda} = \mathbf{L}l(t) \\ s(t) = l(t) = v(t) \end{cases} \tag{4.37}$$

To approximate the search direction equations, a Galerkin method can be used yielding to:

$$\mathbf{L} = \frac{\int_0^T \mathbf{res}_C v(t)\,\mathrm{d}t}{\int_0^T v^2(t)\,\mathrm{d}t} - \mathbf{k}\mathbf{B}\mathbf{V} \qquad \mathbf{S} = \frac{\int_0^T \mathbf{res}_P v(t)\,\mathrm{d}t}{\int_0^T v^2(t)\,\mathrm{d}t} + \mathbf{D}\mathbf{B}_\varepsilon\mathbf{V} \tag{4.38}$$

One can verify that the above modes satisfy conditions in equation (4.35) thanks to (4.36). This is a formulation for the enrichment stage. Future works would consist in formulating a suited preliminary stage and an efficient strategy to store the suggested sets of modes.



(a) t_1 (b) t_{10} (c) t_{20} (d) t_{30} (e) t_{40} (f) t_{50}

Figure 4.6 – Solution **s**: displacement and contact force fields. The color map refers to the cumulated plastic strain and red arrows correspond to nodal contact force.

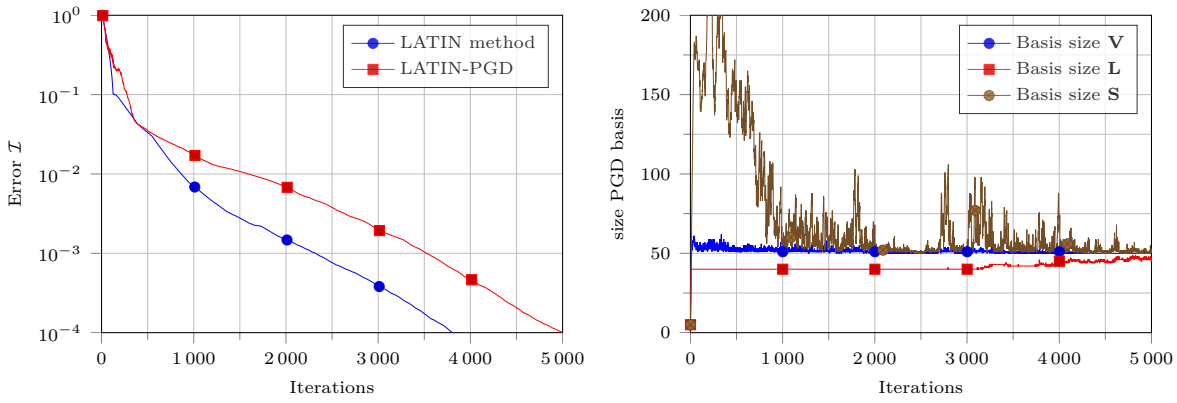
4.2.3 Numerical application

The suggested LATIN formulation with frictional contact for elastoplastic material is applied on the two dimensional extrusion problem. Additional material properties are considered such as the yield stress $\sigma_Y = 3.1 \times 10^7$ and the hardening modulus $h = 2.612 \times 10^8$ ($h = \frac{E}{264}$). For sake of computational time, the time interval is discretized using $\Delta t = 0.02$. Some snapshots of the solution are plotted in figure 4.6 and convergence of the proposed LATIN formulations on figure 4.5.

The number of required iterations to reach a reasonable accuracy is quite large. Moreover, one has to keep in mind that the non-linear stage tackling the plasticity behavior is computationally expensive (whereas the non-linear and local stage tackling the frictional behavior is cheap). That can be firstly explained by the fact that the search direction for the contact behavior are not optimal. Indeed, they depend on the material stiffness (*i.e.* Young modulus) which is softened (*i.e.* hardening modulus) in the plasticity regime. This technical aspect could be a prospect to enhance the suggested algorithm.

4.3 General outlooks

Extension of the PGD method to parametric applications and extension to other kind of non-linearity are direct and pragmatic continuations for work developed in this dissertation. Other orientations could also be envisaged such as large sliding motion, large displacements [BLPR97], transient dynamic and impact [OBG10, LLB00, LBL02, BGA13]. Analogy with multigrid methods may also be further investigated by deploying localized multigrid strategy as in [Ran08].



(a) Convergence of LATIN and PGD

(b) Basis sizes

Figure 4.7 – Convergence of the LATIN method and LATIN-PGD method for elastoplastic material and frictional contact.

Conclusion

In this dissertation, innovative strategies based on model reduction methods were suggested to solve efficiently the non-linear and time-dependent frictional contact problems. It has been illustrated that a frictional solution is expected to be easily compressed. In other words, such a solution can be cast into a separated form or a low rank format using for instance the singular value decomposition. From the provided singular time-space modes, an interesting scale separability phenomenon has been pointed out. The first modes which are the most energetic ones capture the global trend of the solution and are related to the structural scale of the studied body. Then, next ones are related to localized phenomenon such as for instance frictional contact zones or boundary conditions. This observation can be carried out on both spatial and temporal dimensions. Despite the non-smooth properties of the involved constitutive laws, a space-time frictional solution can be written into a separated form and accurately approximated into a reduced subspace. Thus, model reduction techniques can be applied using a global space-time approach.

The core of the presented work is related to the so-called non-linear large time increment (LATIN) method. This method tackles a problem using a non-incremental approach (*i.e.* space-time approach) and iterates on global space-time solutions. In this way, the different scales of the whole space-time solution can be rapidly identified and such a feature would not be possible with classic incremental approaches such as for instance the Newton-Raphson method. Indeed, building incrementally the solution time step per time step does not allow to have knowledge about dominant trends or modes of the whole time-space solution before having the last time steps converged. This is a key point for implementing efficient model reduction strategies.

First, an accelerated LATIN method was proposed: the FAS/LATIN method. Scale separability observations foster multilevel computational techniques such as multigrid method. The FAS/LATIN idea is inspired from the multigrid full approximation scheme consisting in adapting the computational support (grids) to the process wavelength components of the solution. In our case, some modes characterizing different scales of the solution are considered as grids and corrections are cheaply computed in a reduced basis composed of these modes. In this way, LATIN iterations considered as smoothing relaxations can be greatly accelerated. Obtained performances depend strongly on the relevancy of the provided modes. To compute them, several possibilities were suggested such as surrogate models based on coarse space-time discretization (quite cheap compu-

tational work), on-the-fly computed modes (more expensive to achieve), etc. Nonetheless, these requirements for modes are the main drawbacks of the method.

Second, a quasi-optimal LATIN reduced basis strategy was proposed. Such an approach embeds the PGD method and aims at searching directly from scratch (*i.e.* without prior knowledge) the solution into a separated form. So, on the one hand, computational work can be greatly spared because of only univariate functions are tackled. On the other hand, memory storage issues are addressed due to the prescribed low rank format (PGD expansion) for the solution. The main attractive feature of the proposed strategy lies on the so-called quasi-optimality property. Indeed, it has been illustrated that the processed PGD expansion is close to SVD modes of the reference solution. This is achieved by efficient techniques mastering through non-linear space-time iterations the PGD expansion of the solution (extended preliminary stage and downsizing).

Finally, some development paths for further extensions of this work were roughly studied. Indeed, in this dissertation model reduction methods were essentially deployed to develop efficient solvers. But they also address parametric studies. Thus, a LATIN formulation based on reduced order model (ROM) formulation was proposed to solve several problems with a parametrized frictional coefficient. Nonetheless, challenging issues such as building a relevant reduced basis has to be studied. An extended space-time-parameter LATIN-PGD formulation was also suggested for this kind of problem. But, it lacks in computational efficiency due to involved high order tensor. Additionally, it has been pointed out that the framework is able to take into consideration other kinds of non-linearities such as elastoplastic behaviors.

In this dissertation, only academic cases were studied as examples. Of course, the next pragmatic extension of this work would be the implementation of all of those methods on industrial large scale problem including large sliding motion and general multibodies aspects. Such challenges involve numerous issues related to computational geometry (contact detection) and interesting solutions are suggested in [YCF11, AYM13].

Bibliography

[ACCF09] D. Amsallem, J. Cortial, K. Carlberg, and C. Farhat. A method for interpolating on manifolds structural dynamics reduced-order models. *International journal for numerical methods in engineering*, 80(9):1241–1258, 2009.

[ACF10] D. Amsallem, J. Cortial, and C. Farhat. Towards real-time computational-fluid-dynamics-based aeroelastic computations using a database of reduced-order information. *AIAA journal*, 48(9):2029–2037, 2010.

[AD08] P. Alart and D. Dureisseix. A scalable multiscale latin method adapted to nonsmooth discrete media. *Computer Methods in Applied Mechanics and Engineering*, 197(5):319–331, 2008.

[ADR06] P. Alart, D. Dureisseix, and M. Renouf. *Using Nonsmooth Analysis for Numerical Simulation of Contact Mechanics*, volume 12 of *Advances in Mechanics and Mathematics*. Springer US, 2006.

[AF08] D. Amsallem and C. Farhat. Interpolation method for adapting reduced-order models and application to aeroelasticity. *AIAA journal*, 46(7):1803–1813, 2008.

[AF09] P. Avery and C. Farhat. The feti family of domain decomposition methods for inequality-constrained quadratic programming: Application to contact problems with conforming and nonconforming interfaces. *Computer Methods in Applied Mechanics and Engineering*, 198(21):1673–1683, 2009.

[AF11] D. Amsallem and C. Farhat. An online method for interpolating linear parametric reduced-order models. *SIAM Journal on Scientific Computing*, 33(5):2169–2198, 2011.

[AFL07] D. Amsallem, C. Farhat, and T. Lieu. Aeroelastic analysis of f-16 and f-18/a configurations using adapted cfd-based reduced-order models. In *The 48th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, pages 1–20, 2007.

[AL95] P. Alart and F. Lebon. Solution of frictional contact problems using ilu and coarse/fine preconditioners. *Computational mechanics*, 16(2):98–105, 1995.

[Ala97] P. Alart. Méthode de newton généralisée en mécanique du contact. *Journal de mathématiques pures et appliquées*, 76(1):83–108, 1997.

[Ale02] M. Alexa. Recent advances in mesh morphing. In *Computer graphics forum*, volume 21, pages 173–198. Wiley Online Library, 2002.

[All03] R. J. Allemang. The modal assurance criterion–twenty years of use and abuse. *Sound and vibration*, 37(8):14–23, 2003.

[AMCK06] Amine Ammar, B Mokdad, F Chinesta, and Roland Keunings. A new family of solvers for some classes of multidimensional partial differential equations encountered in kinetic theory modeling of complex fluids. *Journal of Non-Newtonian Fluid Mechanics*, 139(3):153–176, 2006.

[AMCK07] A Ammar, B Mokdad, F Chinesta, and Roland Keunings. A new family of solvers for some classes of multidimensional partial differential equations encountered in kinetic theory modelling of complex fluids: Part ii: Transient simulation using space-time separated representations. *Journal of Non-Newtonian Fluid Mechanics*, 144(2):98–121, 2007.

[Amo32] G. Amontons. De la resistance causée dans des machines, tant par les frottements des parties que les composent, que par la roideur des cordes qu'on y employe, et la maniere de calculer l'un e l'autre. *Académie Royale des Sciences, Paris, France, Troiséme (1732) edition*, pages 206–222, 1732.

[AP97] M. Anitescu and F. A. Potra. Formulating dynamic multi-rigid-body contact problems with friction as solvable linear complementarity problems. *Nonlinear Dynamics*, 14(3):231–247, 1997.

[ARLF04] P. Avery, G. Rebel, M. Lesoinne, and C. Farhat. A numerically scalable dual-primal substructuring method for the solution of contact problems — part i: the frictionless case. *Computer Methods in Applied Mechanics and Engineering*, 193(23):2403–2426, 2004.

[AYM13] A. M. Aragón, V. A. Yastrebov, and J. Molinari. A constrained-optimization methodology for the detection phase in contact mechanics simulations. *International Journal for Numerical Methods in Engineering*, 96(5):323–338, 2013.

[AZF12a] D. Amsallem, M. J. Zahr, and C. Farhat. Nonlinear model order reduction based on local reduced-order bases. *International Journal for Numerical Methods in Engineering*, 92(10):891–916, 2012.

[AZF12b] D. Amsallem, M.J. Zahr, and C. Farhat. Nonlinear model order reduction based on local reduced-order bases. *International Journal for Numerical Methods in Engineering*, 92(10):891–916, 2012.

[BAV01] M. Barboteu, P. Alart, and M. Vidrascu. A domain decomposition strategy for nonclassical frictional multi-contact problems. *Computer Methods in Applied Mechanics and Engineering*, 190(37):4785–4803, 2001.

[BB05] P. Ballard and S. Basseville. Existence and uniqueness for dynamical unilateral contact with coulomb friction: a model problem. *ESAIM: Mathematical Modelling and Numerical Analysis*, 39(01):59–77, 2005.

[BBL90] P. Boisse, P. Bussy, and P. Ladevèze. A new approach in non-linear mechanics: The large time increment method. *International journal for numerical methods in engineering*, 29(3):647–663, 1990.

[BC68] M. CC. Bampton and R. R. Craig, Jr. Coupling of substructures for dynamic analyses. *AIAA Journal*, 6(7):1313–1319, 1968.

[BC03] P.-A. Boucard and L. Champaney. A suitable computational strategy for the parametric analysis of problems with multiple contact. *International Journal for Numerical Methods in Engineering*, 57(9):1259–1281, 2003.

[Ber04] M. Bergmann. *Optimisation aérodynamique par réduction de modèle POD et contrôle optimal: application au sillage laminaire d'un cylindre circulaire*. PhD thesis, Vandoeuvre-les-Nancy, INPL, 2004.

[BF95] D. Boukari and A.-V. Fiacco. Survey of penalty, exact-penalty and multiplier methods from 1968 to 1993*. *Optimization*, 32(4):301–334, 1995.

[BGA13] L. Boucinha, A. Gravouil, and A. Ammar. Space-time proper generalized decompositions for the resolution of transient elastodynamic models. *Computer Methods in Applied Mechanics and Engineering*, 255:67 – 88, 2013.

[BGH00] G. J. Barclay, D. F. Griffiths, and D. J. Higham. Theta method dynamics. *LMS Journal of Computation and Mathematics*, 3:27–43, 2000.

[BGLC10] E. Biotteau, A. Gravouil, A. A. Lubrecht, and A. Combescure. Multigrid solver with automatic mesh refinement for transient elastoplastic dynamic problems. *International Journal for Numerical Methods in Engineering*, 84(8):947–971, 2010.

[BGVD12] C. G. Baker, K. A Gallivan, and P. Van Dooren. Low-rank incremental methods for computing dominant singular subspaces. *Linear Algebra and its Applications*, 436(8):2866–2888, 2012.

[BHL98] F. B. Belgacem, P. Hild, and P. Laborde. The mortar finite element method for contact problems. *Mathematical and computer modelling*, 28(4):263–271, 1998.

[BHM00] W. Briggs, V. Henson, and S. McCormick. *A Multigrid tutorial 2nd ed.* Siam, Philadelphia, 2000.

[BHS03] Roland Becker, Peter Hansbo, and Rolf Stenberg. A finite element method for domain decomposition with non-matching grids. *ESAIM: Mathematical Modelling and Numerical Analysis*, 37(02):209–225, 2003.

[BL11] Achi Brandt and Oren E Livne. *Multigrid techniques: 1984 guide with applications to fluid dynamics*, volume 67. SIAM, 2011.

[BLPR97] P.-A. Boucard, P. Ladevèze, M. Poss, and P. Rougée. A nonincremental approach for large displacement problems. *Computers & structures*, 64(1):499–508, 1997.

[BM00] C. Bernardi and Y. Maday. Mesh adaptivity in finite elements using the mortar method. *Revue européenne des éléments finis*, 9(4):451–465, 2000.

[BMH85] A Brandt, S McCoruick, and J Huge. Algebraic multigrid (amg) f0r sparse matrix equati0ns. *Sparsity and its Applications*, page 257, 1985.

[BMP93] Christine Bernardi, Yvon Maday, and Anthony T Patera. Domain decomposition by the mortar element method. In *Asymptotic and numerical methods for partial differential equations with critical parameters*, pages 269–286. Springer, 1993.

[BMP12] P. Bussetta, D. Marceau, and J.-P. Ponthot. The adapted augmented lagrangian method: a new method for the resolution of the mechanical frictional contact problem. *Computational Mechanics*, 49(2):259–275, 2012.

[BN78] J. R. Bunch and C. P. Nielsen. Updating the singular value decomposition. *Numerische Mathematik*, 31(2):111–129, 1978.

[Bra77] A. Brandt. Multi-level adaptive solutions to boundary-value problems. *Mathematics of computation*, 31(138):333–390, 1977.

[Bra02] M. Brand. Incremental singular value decomposition of uncertain data with missing values. In *Computer Vision–ECCV 2002*, pages 707–720. Springer, 2002.

[Bra05] A. Brandt. Multiscale solvers and systematic upscaling in computational physics. *Computer Physics Communications*, 169:438 – 441, 2005.

[Bra06] M. Brand. Fast low-rank modifications of the thin singular value decomposition. *Linear algebra and its applications*, 415(1):20–30, 2006.

[Bro97] R. Bro. Parafac. tutorial and applications. *Chemometrics and intelligent laboratory systems*, 38(2):149–171, 1997.

[CA88] A. Curnier and P. Alart. A generalized newton method for contact problems with friction. *Journal de mécanique théorique et appliquée*, 7:67–82, 1988.

[CAC10] F. Chinesta, A. Ammar, and E. Cueto. Recent advances and new challenges in the use of the proper generalized decomposition for solving multidimensional models. *Archives of Computational methods in Engineering*, 17(4):327–350, 2010.

[CALK11] F. Chinesta, A. Ammar, A. Leygue, and R. Keunings. An overview of the proper generalized decomposition with applications in computational rheology. *Journal of Non-Newtonian Fluid Mechanics*, 166(11):578–592, 2011.

[Cap11] A. Capatina. Inéquations variationnelles et problèmes de contact avec frottement. *Preprint Series IMAR*, 10:1–210, 2011.

[CC14] Francisco Chinesta and Elías Cueto. *PGD-Based Modeling of Materials, Structures and Processes*. Springer, 2014.

[CCDL97] L. Champaney, J.-Y. Cognard, D. Dureisseix, and P. Ladevèze. Large scale applications on parallel computers of a mixed domain decomposition method. *Computational Mechanics*, 19:253–263, 1997.

[CCL99] L. Champaney, J.-Y. Cognard, and P. Ladevèze. Modular analysis of assemblages of three-dimensional structures with unilateral contact conditions. *Computers and Structures*, 73:249–266, 1999.

[CDJGK03] P. Colella, T. H. Dunning Jr., W. D. Gropp, and D. E. Keyes. A science-based case for large-scale simulation. *Washington, DC: DOE Office of Science*, 2003.

[CFCA13] K. Carlberg, C. Farhat, J. Cortial, and D. Amsallem. The gnat method for nonlinear model reduction: effective implementation and application to computational fluid dynamics and turbulent flows. *Journal of Computational Physics*, 242:623–647, 2013.

[CH93] J. Chung and G.M. Hulbert. A time integration algorithm for structural dynamics with improved numerical dissipation: the generalized-α method. *Journal of applied mechanics*, 60(2):371–375, 1993.

[Cha96] Laurent Champaney. *Une nouvelle approche modulaire pour l'analyse d'assemblages de structures tridimensionnelles*. PhD thesis, École normale supérieure de Cachan-ENS Cachan, 1996.

[CKL14] F. Chinesta, R. Keunings, and A. Leygue. *The Proper Generalized Decomposition for Advanced Numerical Simulations*. Springer International Publishing, 2014.

[CL93] J.-Y. Cognard and P. Ladevèze. A large time increment approach for cyclic viscoplasticity. *International journal of plasticity*, 9(2):141–157, 1993.

[CLB+13] F. Chinesta, A. Leygue, F. Bordeu, J.V. Aguado, E. Cueto, D. Gonzalez, I. Alfaro, A. Ammar, and A. Huerta. Pgd-based computational vademecum for efficient design, optimization and control. *Archives of Computational Methods in Engineering*, 20(1):31–59, 2013.

[CLC11] F. Chinesta, P. Ladevèze, and E. Cueto. A short review on model order reduction based on proper generalized decomposition. *Archives of Computational Methods in Engineering*, 18:395–404, 2011.

[Coc84] Marius Cocu. Existence of solutions of signorini problems with friction. *International journal of engineering science*, 22(5):567–575, 1984.

[Cou85] C.-A. Coulomb. *Théorie des machines simples*. Mémoires de l'Académie des Sciences, 1785.

[CPS09] R. W. Cottle, J.-S. Pang, and R. E. Stone. *The linear complementarity problem*, volume 60. Siam, 2009.

[DB06] D. Dureisseix and H. Bavestrello. Information transfer between incompatible finite element meshes: application to coupled thermo-viscoelasticity. *Computer methods in applied mechanics and engineering*, 195(44):6523–6541, 2006.

[DF01] D. Dureisseix and C. Farhat. A numerically scalable domain decomposition method for the solution of frictionless contact problems. *International Journal for Numerical Methods in Engineering*, 50(12):2643–2666, 2001.

[DHK02] Z. Dostál, J. Haslinger, and R. Kučera. Implementation of the fixed point method in contact problems with coulomb friction based on a dual splitting type technique. *Journal of computational and applied mathematics*, 140(1):245–256, 2002.

[DHK+05] Z. Dostál, D. Horák, R. Kučera, V. Vondrák, J. Haslinger, J. Dobiáš, and S. Pták. Feti based algorithms for contact problems: scalability, large

displacements and 3d coulomb friction. *Computer Methods in Applied Mechanics and Engineering*, 194(2):395–409, 2005.

[DKH+10] Zdenek Dostál, Tomás Kozubek, Petr Horyl, Tomás Brzobohatỳ, and Alexandros Markopoulos. A scalable tfeti algorithm for two-dimensional multibody contact problems with friction. *Journal of computational and applied mathematics*, 235(2):403–418, 2010.

[DKM+12] Zdeněk Dostál, Tomáš Kozubek, Alexandros Markopoulos, Tomáš Brzobohatỳ, Vít Vondrák, and Petr Horyl. A theoretically supported scalable tfeti algorithm for the solution of multibody 3d contact problems with friction. *Computer Methods in Applied Mechanics and Engineering*, 205:110–120, 2012.

[DLDMV00a] L. De Lathauwer, B. De Moor, and J. Vandewalle. A multilinear singular value decomposition. *SIAM journal on Matrix Analysis and Applications*, 21(4):1253–1278, 2000.

[DLDMV00b] L. De Lathauwer, B. De Moor, and J. Vandewalle. On the best rank-1 and rank-(r 1, r 2,..., rn) approximation of higher-order tensors. *SIAM Journal on Matrix Analysis and Applications*, 21(4):1324–1342, 2000.

[DMB01] John Dolbow, Nicolas Moës, and Ted Belytschko. An extended finite element method for modeling crack growth with frictional contact. *Computer Methods in Applied Mechanics and Engineering*, 190(51):6825–6846, 2001.

[Dru57] D. C. Drucker. A definition of stable inelastic material. Technical report, DTIC Document, 1957.

[DS05] Z. Dostál and J. Schoberl. Minimizing quadratic functions subject to bound constraints with the rate of convergence and finite termination. *Computational Optimization and Applications*, 30(1):23–43, 2005.

[DSF98] G De Saxcé and Z-Q Feng. The bipotential method: a constructive approach to design the complete contact law with friction and improved numerical algorithms. *Mathematical and Computer Modelling*, 28(4):225–245, 1998.

[Dur97] D. Dureisseix. *Une approche multi-échelles pour des calculs de structures sur ordinateurs à architecture parallèle*. PhD thesis, École normale supérieure de Cachan-ENS Cachan, 1997.

[EAB01] Nagi El-Abbasi and Klaus-Jürgen Bathe. Stability and patch test performance of contact discretizations and a new solution algorithm. *Computers & Structures*, 79(16):1473–1486, 2001.

[Ern04] A. Ern. *Theory and practice of finite elements*, volume 159. Springer, 2004.

[EY36] C. Eckart and G. Young. The approximation of one matrix by another of lower rank. *Psychometrika*, 1:211–218, 1936.

[FG00] M. Fortin and R. Glowinski. *Augmented Lagrangian methods: applications to the numerical solution of boundary-value problems*. Access Online via Elsevier, 2000.

[FHW04] Achim Fritz, Stefan Hüeber, and Barbara I Wohlmuth. A comparison of mortar and nitsche techniques for linear elasticity. *Calcolo*, 41(3):115–137, 2004.

[Fic64] G. Fichera. *Elastostatics problems with unilateral constraints: the Signorini problem with ambiguous boundary conditions*. Aerospace Research Laboratories, 1964.

[Fic73] G. Fichera. Boundary value problems of elasticity with unilateral constraints. In *Linear Theories of Elasticity and Thermoelasticity*, pages 391–424. Springer, 1973.

[FZ75] A Francavilla and OC Zienkiewicz. A note on numerical computation of elastic contact problems. *International Journal for Numerical Methods in Engineering*, 9(4):913–924, 1975.

[Gal11] F. Galland. *An adaptive model reduction approach for 3D fatigue crack growth in small scale yielding conditions*. PhD thesis, INSA de Lyon, 2011.

[GDGR14] A. Giacoma, D. Dureisseix, A. Gravouil, and M. Rochette. A multiscale large time increment/fas algorithm with time-space model reduction for frictional contact problems. *International Journal for Numerical Methods in Engineering*, 97(3):207–230, 2014.

[GGMR11] F. Galland, A. Gravouil, E. Malvesin, and M. Rochette. A global model reduction approach for 3d fatigue crack growth with confined plasticity. *Computer Methods in Applied Mechanics and Engineering*, 200(5):699–716, 2011.

[GJMN05] M. J. Gander, C. Japhet, Y. Maday, and F. Nataf. A new cement to glue nonconforming grids with robin interface conditions: The finite element case. In *Domain decomposition methods in science and engineering*, pages 259–266. Springer, 2005.

[GKC+09] S. Glotzer, S. Kim, P. T. Cummings, A. Deshmukh, M. Head-Gordon, G. Karniadakis, L. Petzold, C. Sagui, and M. Shinozuka. International

assessment of research and development in simulation-based engineering and science. *World Technology Evaluation Center, Maryland*, 2009.

[GLMN14] L. Giraldi, D. Liu, H. G. Matthies, and A. Nouy. To be or not to be intrusive? the solution of parametric and stochastic equations—proper generalized decomposition. *arXiv preprint arXiv:1405.0875*, 2014.

[GR70] G. H. Golub and C. Reinsch. Singular value decomposition and least squares solutions. *Numerische Mathematik*, 14(5):403–420, 1970.

[Gre95] L. Grego. *Méthodes multiniveaux pour des problèmes de contact unilatéral avec frottement*. PhD thesis, 1995.

[Gri21] Alan A Griffith. The phenomena of rupture and flow in solids. *Philosophical transactions of the royal society of london. Series A, containing papers of a mathematical or physical character*, pages 163–198, 1921.

[GVL12] G. H. Golub and C. F. Van Loan. *Matrix computations*, volume 3. JHU Press, 2012.

[HBN12] C. Heyberger, P.-A. Boucard, and D. Néron. Multiparametric analysis within the proper generalized decomposition framework. *Computational Mechanics*, 49(3):277–289, 2012.

[HBN13] C. Heyberger, P.-A. Boucard, and D. Néron. A rational strategy for the resolution of parametrized problems in the PGD framework. *Computer Methods in Applied Mechanics and Engineering*, 259:40–49, 2013.

[Hil58] R. Hill. A general theory of uniqueness and stability in elastic-plastic solids. *Journal of the Mechanics and Physics of Solids*, 6(3):236–249, 1958.

[Hil00] P. Hild. Numerical implementation of two nonconforming finite element methods for unilateral contact. *Computer Methods in Applied Mechanics and Engineering*, 184(1):99–123, 2000.

[HL99] M. W. Heinstein and T. A. Laursen. An algorithm for the matrix-free solution of quasistatic frictional contact problems. *International Journal for Numerical Methods in Engineering*, 44(9):1205–1226, 1999.

[HL03] M.W. Heinstein and T.A. Laursen. A three dimensional surface-to-surface projection algorithm for non-coincident domains. *Communications in numerical methods in engineering*, 19(6):421–432, 2003.

[HLB98] P. Holmes, J. L Lumley, and G. Berkooz. *Turbulence, coherent structures, dynamical systems and symmetry*. Cambridge university press, 1998.

[Hor10] MF Horstemeyer. Multiscale modeling: a review. In *Practical aspects of computational chemistry*, pages 87–135. Springer, 2010.

[HSW12] B. Haasdonk, J. Salomon, and B. Wohlmuth. A reduced basis method for parametrized variational inequalities. *SIAM Journal on Numerical Analysis*, 50(5):2656–2676, 2012.

[HT96] W. Hu and P.F. Thomson. An evaluation of a large time increment method. *Computers & structures*, 58(3):633–637, 1996.

[HTS+76] Thomas JR Hughes, Robert L Taylor, Jerome L Sackman, Alain Curnier, and Worsak Kanoknukulchai. A finite element method for a class of contact-impact problems. *Computer Methods in Applied Mechanics and Engineering*, 8(3):249–276, 1976.

[HUL01] J.-B. Hiriart-Urruty and C. Lemaréchal. *Fundamentals of convex analysis*. Springer, 2001.

[JAJ98] F. Jourdan, P. Alart, and M. Jean. A gauss-seidel like algorithm to solve frictional contact problems. *Computer methods in applied mechanics and engineering*, 155(1):31–47, 1998.

[Jea99] M. Jean. The non-smooth contact dynamics method. *Computer methods in applied mechanics and engineering*, 177(3):235–257, 1999.

[JF08] Pierre Joli and Z-Q Feng. Uzawa and newton algorithms to solve frictional contact problems within the bi-potential framework. *International journal for numerical methods in engineering*, 73(3):317–330, 2008.

[KB09] T. G. Kolda and B. W. Bader. Tensor decompositions and applications. *SIAM review*, 51(3):455–500, 2009.

[KGAB11] P. Kerfriden, P. Gosselet, S. Adhikari, and S.P.A. Bordas. Bridging proper orthogonal decomposition methods and augmented newton-krylov algorithms: An adaptive model order reduction for highly nonlinear mechanical problems. *Computer Methods in Applied Mechanics and Engineering*, 200:850 – 866, 2011.

[Kla90] A. Klarbring. Examples of non-uniqueness and non-existence of solutions to quasistatic contact problems with friction. *Ingenieur-Archiv*, 60(8):529–541, 1990.

[KO88] N. Kikuchi and J. T. Oden. *Contact problems in elasticity: a study of variational inequalities and finite element methods*, volume 8. siam, 1988.

[Kol01] T. G. Kolda. Orthogonal tensor decompositions. *SIAM Journal on Matrix Analysis and Applications*, 23(1):243–255, 2001.

[KP14] J. Krenciszek and R. Pinnau. Model reduction of contact problems in elasticity: Proper orthogonal decomposition for variational inequalities. In *Progress in Industrial Mathematics at ECMI 2012*, pages 277–284. Springer, 2014.

[KPB11] P. Kerfriden, J.-C. Passieux, and S.P.A. Bordas. Local/global model order reduction strategy for the simulation of quasi-brittle fracture. *International Journal for Numerical methods in Engineering*, 89:154–179, 2011.

[KV01] K. Kunisch and S. Volkwein. Galerkin proper orthogonal decomposition methods for parabolic problems. *Numerische Mathematik*, 90:117–148, 2001.

[Lad99] P. Ladevèze. *Nonlinear Computational Structural Methods: New Approaches and Non-Incremental Methods of Calculation*. Springer-Verlag New York Inc., 1999.

[Lau92a] T. A. Laursen. *Formulation and treatment of frictional contact problems using finite elements*. PhD thesis, Stanford University, 1992.

[Lau92b] T. A. Laursen. *Formulation and treatment of frictional contact problems using finite elements*. PhD thesis, Stanford University, CA, 1992.

[LBL02] H. Lemoussu, P.-A. Boucard, and P. Ladevèze. A 3D shock computational strategy for real assembly and shock attenuator. *Advances in Engineering Software*, 33:517 – 526, 2002.

[LC11] P. Ladevèze and L. Chamoin. On the verification of model reduction methods based on the proper generalized decomposition. *Computer Methods in Applied Mechanics and Engineering*, 200(23):2032–2047, 2011.

[LDH11] D. Luo, C. Ding, and H. Huang. Are tensor decomposition solutions unique? on the global convergence hosvd and parafac algorithms. In *Advances in Knowledge Discovery and Data Mining*, pages 148–159. Springer, 2011.

[Leb89] F. Lebon. *Résolution numérique de problèmes de frottement de Coulomb. Accélération de convergence par une méthode multigrilles interne*. PhD thesis, Université de Provence, 1989.

[LF00] D. Li and M. Fukushima. Smoothing newton and quasi-newton methods for mixed complementarity problems. *Computational Optimization and Applications*, 17(2-3):203–230, 2000.

[LLB00] P. Ladevèze, H. Lemoussu, and P.-A. Boucard. A modular approach to 3-D impact computation with frictional contact. *Computers and Structures*, 78:45 – 51, 2000.

[LM97] Catherine Lacour and Yvon Maday. Two different approaches for matching nonconforming grids: The mortar element method and the feti method. *BIT Numerical Mathematics*, 37(3):720–738, 1997.

[LP05] P. Ladevèze and J.-P. Pelle. *Mastering calculations in linear and nonlinear mechanics*. Springer, 2005.

[LPN10] P. Ladevèze, J.-C. Passieux, and D. Néron. The LATIN multiscale computational method and the proper generalized decomposition. *Computer Methods in Applied Mechanics and Engineering*, 199:1287 – 1296, 2010.

[LR10] T. Lassila and G. Rozza. Parametric free-form shape design with pde models and reduced basis method. *Computer Methods in Applied Mechanics and Engineering*, 199(23):1583–1592, 2010.

[LRLG91] F. Lebon, M. Raous, J.-C. Latil, and L. Grego. Multigrid method in non linear structure mechanics. *EUROPEAN CONFERENCE ON NEW ADVANCES IN COMPUTATIONAL STRUCTURAL MECHANICS*, 1991.

[LRR07] F. Lebon, M. Raous, and I. Rosu. Multigrid methods for unilateral contact problems with friction. In *IUTAM Symposium on Computational Methods in Contact Mechanics*, volume 3 of *IUTAM Bookseries*, pages 1–16. 2007.

[LT94] P. Le Tallec. Domain decomposition methods in computational mechanics. *Computational mechanics advances*, 1(2):121–220, 1994.

[LTS95] P. Le Tallec and T. Sassi. Domain decomposition with nonmatching grids: augmented lagrangian approach. *Mathematics of Computation*, 64(212):1367–1396, 1995.

[Man66] J. Mandel. Conditions de stabilité et postulat de drucker. In *Rheology and Soil Mechanics/Rhéologie et Mécanique des Sols*, pages 58–68. Springer, 1966.

[MDL88] O. L. Mangasarian and R. De Leone. Parallel gradient projection successive overrelaxation for symmetric linear complementarity problems and linear programs. *Annals of Operations Research*, 14(1):41–59, 1988.

[ML00] TW McDevitt and TA Laursen. A mortar-finite element formulation for frictional contact problems. *International Journal for Numerical Methods in Engineering*, 48(10):1525–1547, 2000.

[MNPP07] Y. Maday, N. C. Nguyen, A. T. Patera, and G. S.H. Pau. A general, multipurpose interpolation procedure: the magic points. 2007.

[Moo65] G. E. Moore. Cramming more components onto integrated circuits, 1965.

[Mor67] J.-J. Moreau. Fonctionnelles convexes. *Séminaire Jean Leray*, (2):1–108, 1967.

[Mor76] J.-J. Moreau. Application of convex analysis to the treatment of elastoplastic systems. In *Applications of methods of functional analysis to problems in mechanics*, pages 56–89. Springer, 1976.

[Mor88] J.-J. Moreau. Unilateral contact and dry friction in finite freedom dynamics. In *Nonsmooth mechanics and Applications*, pages 1–82. Springer, 1988.

[NACC08] S. Niroomandi, I. Alfaro, E. Cueto, and F. Chinesta. Real-time deformable models of non-linear tissues by model reduction techniques. *Computer Methods and Programs in Biomedicine*, 91(3):223–231, 2008.

[Nou10] A. Nouy. A priori model reduction through proper generalized decomposition for solving time-dependent partial differential equations. *Computer Methods in Applied Mechanics and Engineering*, 199(23):1603–1626, 2010.

[OBG10] D. Odièvre, P.-A. Boucard, and F. Gatuingt. A parallel, multiscale domain decomposition method for the transient dynamic analysis of assemblies with friction. *Computer Methods in Applied Mechanics and Engineering*, 199(21):1297–1306, 2010.

[OS86] M. Ortiz and J.-C. Simo. An analysis of a new class of integration algorithms for elastoplastic constitutive relations. *International Journal for Numerical Methods in Engineering*, 23(3):353–366, 1986.

[PBG10] E. Pierres, M.-C. Baietto, and A. Gravouil. A two-scale extended finite element method for modelling 3D crack growth with interfacial contact. *Computer Methods in Applied Mechanics and Engineering*, 199:1165 – 1177, 2010.

[PC99] G. Pietrzak and A. Curnier. Large deformation frictional contact mechanics: continuum formulation and augmented lagrangian treatment. *Computer Methods in Applied Mechanics and Engineering*, 177(3):351–381, 1999.

[PL04] Michael A Puso and Tod A Laursen. A mortar segment-to-segment contact method for large deformation solid mechanics. *Computer Methods in Applied Mechanics and Engineering*, 193(6):601–629, 2004.

[Pon98] J.-P. Ponthot. An extension of the radial return algorithm to account for rate-dependent effects in frictional contact and visco-plasticity. *Journal of Materials Processing Technology*, 80:628–634, 1998.

[Pra49] W. Prager. Recent developments in the mathematical theory of plasticity. *Journal of Applied Physics*, 20(3):235–241, 1949.

[RA05] M. Renouf and P. Alart. Conjugate gradient type algorithms for frictional multi-contact problems: applications to granular materials. *Computer Methods in Applied Mechanics and Engineering*, 194(18):2019–2041, 2005.

[Ran08] J. Rannou. *Prise en compte d'effets d'échelle en mécanique de la rupture tridimensionnelle par une approche X-FEM multigrille localisée non-linéaire*. PhD thesis, INSA de Lyon, 2008.

[RBDG07] R. Ribeaucourt, M.-C. Baietto-Dubourg, and A. Gravouil. A new fatigue frictional contact crack propagation model with the coupled X-FEM/LATIN method. *Computer Methods in Applied Mechanics and Engineering*, 196:3230 – 3247, 2007.

[RC01] Rémi Rocca and Marius Cocu. Existence and approximation of a solution to quasistatic signorini problem with local friction. *International journal of engineering science*, 39(11):1233–1255, 2001.

[RCB11] V Roulet, L Champaney, and P-A Boucard. A parallel strategy for the multiparametric analysis of structures with large contact and friction surfaces. *Advances in Engineering Software*, 42(6):347–358, 2011.

[RCCA06] D. Ryckelynck, F. Chinesta, E. Cueto, and A. Ammar. On the a priori model reduction: Overview and recent developments. *Archives of Computational Methods in Engineering*, 13:91–128, 2006.

[Ren12] Y. Renard. Generalized newton's methods for the approximation and resolution of frictional contact problems in elasticity. *Computer Methods in Applied Mechanics and Engineering*, 256:38–55, 2012.

[RHP08] Gianluigi Rozza, DBP Huynh, and Anthony T Patera. Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations. *Archives of Computational Methods in Engineering*, 15(3):229–275, 2008.

[RLM11] G. Rozza, T. Lassila, and A. Manzoni. Reduced basis approximation for shape optimization in thermal flows with a parametrized polynomial geometric map. In *Spectral and high order methods for partial differential equations*, pages 307–315. Springer, 2011.

[RM10] Gianluigi Rozza and Andrea Manzoni. Model order reduction by geometrical parametrization for shape optimization in computational fluid dynamics. In *Proceedings of the ECCOMAS CFD 2010, V European Conference on Computational Fluid Dynamics*, number EPFL-CONF-148535, 2010.

[RNB13] N. Relun, D. Néron, and P.-A. Boucard. A model reduction technique based on the pgd for elastic-viscoplastic computational analysis. *Computational Mechanics*, 51(1):83–92, 2013.

[Roc97] R. T. Rockafellar. *Convex analysis*. Number 28. Princeton university press, 1997.

[Ryc05] D. Ryckelynck. A priori hyperreduction method: an adaptive approach. *Journal of Computational Physics*, 202(1):346–366, 2005.

[SC] Alberto Salvadori and Angelo Carini. Variational formulations in fracture mechanics due to plasticity analogies.

[SC11] A Salvadori and A Carini. Minimum theorems in incremental linear elastic fracture mechanics. *International Journal of Solids and Structures*, 48(9):1362–1369, 2011.

[SH] J.C. Simo and T.J.R. Hughes. Computational inelasticity. 1998. *Springer, New York*.

[Sig59] A. Signorini. Questioni di elasticità non linearizzata e semilinearizzata. *Rendiconti di Matematica e delle sue applicazioni, V. Ser.*, 18:95–139, 1959.

[SL92] J.-C. Simo and T. A. Laursen. An augmented lagrangian treatment of contact problems involving friction. *Computers & Structures*, 42(1):97–116, 1992.

[SP91] A.M. Stuart and A.T. Peplow. The dynamics of the theta method. *SIAM journal on scientific and statistical computing*, 12(6):1351–1372, 1991.

[Ste08] M. O. Steinhauser. *Computational Multiscale Modeling of Fluids and Solids*, volume 60. Springer, 2008.

[Stü01] Klaus Stüben. A review of algebraic multigrid. *Journal of Computational and Applied Mathematics*, 128(1):281–309, 2001.

[TGBNT12] B. Trollé, A. Gravouil, M.-C. Baietto, and T.M.L. Nguyen-Tajan. Optimization of a stabilized X-FEM formulation for frictional cracks. *Finite Elements in Analysis and Design*, 59:18 – 27, 2012.

[TP91] Robert L Taylor and Panayiotis Papadopoulos. On a patch test for contact problems in two dimensions. *Computational methods in nonlinear mechanics*, pages 690–702, 1991.

[VL00] C. H. Venner and A. A. Lubrecht. *Multilevel Methods in Lubrication*. Elsevier, 2000.

[VMB96] Petr Vaněk, Jan Mandel, and Marian Brezina. Algebraic multigrid by smoothed aggregation for second and fourth order elliptic problems. *Computing*, 56(3):179–196, 1996.

[Woh00] Barbara I Wohlmuth. A mortar finite element method using dual spaces for the lagrange multiplier. *SIAM Journal on Numerical Analysis*, 38(3):989–1012, 2000.

[Woh11] Barbara Wohlmuth. Variationally consistent discretization schemes and numerical algorithms for contact problems. *Acta Numerica*, 20:569–734, 2011.

[Wri95] P. Wriggers. Finite element algorithms for contact problems. *Archives of Computational Methods in Engineering*, 2:1–49, 1995.

[WS85] P Wriggers and JC Simo. A note on tangent stiffness for fully nonlinear contact problems. *Communications in Applied Numerical Methods*, 1(5):199–203, 1985.

[WZ08] P Wriggers and Giorgio Zavarise. A formulation for frictionless contact problems using a weak form introduced by nitsche. *Computational Mechanics*, 41(3):407–420, 2008.

[Yas13] Vladislav A Yastrebov. *Numerical methods in contact mechanics*. John Wiley & Sons, 2013.

[YCF11] V. A. Yastrebov, G. Cailletaud, and F. Feyel. A local contact detection technique for very large contact and self-contact problems: sequential and parallel implementations. In *Trends in Computational Contact Mechanics*, pages 227–251. Springer, 2011.

[ZDL09] Giorgio Zavarise and Laura De Lorenzis. A modified node-to-segment algorithm passing the contact patch test. *International journal for numerical methods in engineering*, 79(4):379–416, 2009.

Appendix A

Toward the SVD using a recursive compression function

A.1 Compression function F

In the following, we consider the canonical euclidean inner product. Let $\check{\mathbf{u}}$ be the low rank approximation defined as follows:

$$\check{\mathbf{u}} = \mathbf{V}_1\boldsymbol{\phi}_1^T + \mathbf{V}_2\boldsymbol{\phi}_2^T \quad \text{with} \quad \begin{cases} \mathbf{V}_i^T\mathbf{V}_j = \boldsymbol{\delta}_{ij} \\ \|\boldsymbol{\phi}_1\|^2 \geqslant \|\boldsymbol{\phi}_2\|^2 \end{cases} \tag{A.1}$$

We define the function F providing new vectors $\tilde{\mathbf{V}}_1$, $\tilde{\boldsymbol{\phi}}_1$, $\tilde{\mathbf{V}}_2$, $\tilde{\boldsymbol{\phi}}_2$ such that:

$$(\tilde{\mathbf{V}}_1, \tilde{\boldsymbol{\phi}}_1, \tilde{\mathbf{V}}_2, \tilde{\boldsymbol{\phi}}_2) = F(\mathbf{V}_1, \boldsymbol{\phi}_1, \mathbf{V}_2, \boldsymbol{\phi}_2) \tag{A.2}$$

Such a function consists in doing the following steps:

1. Time mode $\boldsymbol{\phi}_2$ is written as follows $\boldsymbol{\phi}_2 = \alpha\boldsymbol{\phi}_1 + \bar{\boldsymbol{\phi}}_2$ with $\boldsymbol{\phi}_1^T\bar{\boldsymbol{\phi}}_2 = 0$ and $\alpha = \frac{\boldsymbol{\phi}_1^T\boldsymbol{\phi}_2}{\boldsymbol{\phi}_1^T\boldsymbol{\phi}_1}$.
2. $\check{\mathbf{u}}$ can be rewritten as follows $\check{\mathbf{u}} = \bar{\mathbf{V}}_1\boldsymbol{\phi}_1^T + \mathbf{V}_2\bar{\boldsymbol{\phi}}_2^T$ with $\bar{\mathbf{V}}_1 = \mathbf{V}_1 + \alpha\mathbf{V}_2$. One can remark that $\bar{\mathbf{V}}_1^T\mathbf{V}_2 \neq 0$.
3. Space modes are reorthogonalized by writting $\mathbf{V}_2 = \beta\bar{\mathbf{V}}_1 + \bar{\mathbf{V}}_2$ with $\bar{\mathbf{V}}_1^T\bar{\mathbf{V}}_2 = 0$ and $\beta = \frac{\bar{\mathbf{V}}_1^T\bar{\mathbf{V}}_2}{\bar{\mathbf{V}}_1^T\bar{\mathbf{V}}_1} = \frac{\alpha}{1+\alpha^2}$.
4. $\check{\mathbf{u}}$ can be rewritten as follows $\check{\mathbf{u}} = \bar{\mathbf{V}}_1\bar{\boldsymbol{\phi}}_1^T + \bar{\mathbf{V}}_2\bar{\boldsymbol{\phi}}_2^T$ with $\bar{\boldsymbol{\phi}}_1 = \boldsymbol{\phi}_1 + \beta\bar{\boldsymbol{\phi}}_2$.
5. Space modes are unitarized by doing $\begin{cases} \tilde{\mathbf{V}}_1 = \bar{\mathbf{V}}_1/\|\bar{\mathbf{V}}_1\| \\ \tilde{\boldsymbol{\phi}}_1 = \|\bar{\mathbf{V}}_1\|\bar{\boldsymbol{\phi}}_1 \end{cases}$ and $\begin{cases} \tilde{\mathbf{V}}_2 = \bar{\mathbf{V}}_2/\|\bar{\mathbf{V}}_2\| \\ \tilde{\boldsymbol{\phi}}_2 = \|\bar{\mathbf{V}}_2\|\bar{\boldsymbol{\phi}}_2 \end{cases}$.

Due to the fact that new vectors are obtained by input vector partitions, $\check{\mathbf{u}}$ can be rewritten exactly as follows:

$$\check{\mathbf{u}} = \tilde{\mathbf{V}}_1\tilde{\boldsymbol{\phi}}_1^T + \tilde{\mathbf{V}}_2\tilde{\boldsymbol{\phi}}_2^T \tag{A.3}$$

with the following properties:

$$\tilde{\mathbf{V}}_i^T \tilde{\mathbf{V}}_j = \boldsymbol{\delta}_{ij} \quad \text{Orthogonality of left vectors} \tag{A.4}$$

$$\boldsymbol{\phi}_1^T \boldsymbol{\phi}_2 \geqslant \tilde{\boldsymbol{\phi}}_1^T \tilde{\boldsymbol{\phi}}_2 \quad \text{Compression} \tag{A.5}$$

$$\|\tilde{\boldsymbol{\phi}}_1\|^2 \geqslant \|\tilde{\boldsymbol{\phi}}_2\|^2 \quad \text{Sorting by ascending order} \tag{A.6}$$

In this chapter, we first provide a proof for properties (A.4), (A.5), (A.6) and then we will generalize the function F to a recursive procedure computing the SVD of $\check{\mathbf{u}}$. We recalls the following useful formulae which will be used in the following.

$$\alpha = \frac{\boldsymbol{\phi}_1^T \boldsymbol{\phi}_2}{\boldsymbol{\phi}_1^T \boldsymbol{\phi}_1} \tag{A.7}$$

$$\bar{\boldsymbol{\phi}}_2 = \boldsymbol{\phi}_2 - \alpha \boldsymbol{\phi}_1 \tag{A.8}$$

$$\bar{\mathbf{V}}_1 = \mathbf{V}_1 + \alpha \mathbf{V}_2 \tag{A.9}$$

$$\bar{\boldsymbol{\phi}}_1 = \boldsymbol{\phi}_1 + \frac{\alpha}{1+\alpha^2}(\boldsymbol{\phi}_2 - \alpha \boldsymbol{\phi}_1) = \frac{1}{1+\alpha^2}\boldsymbol{\phi}_1 + \frac{\alpha}{1+\alpha^2}\boldsymbol{\phi}_2 \tag{A.10}$$

$$\bar{\mathbf{V}}_2 = \mathbf{V}_2 - \frac{\alpha}{1+\alpha^2}(\mathbf{V}_1 + \alpha \mathbf{V}_2) = -\frac{\alpha}{1+\alpha^2}\mathbf{V}_1 + \frac{1}{1+\alpha^2}\mathbf{V}_2 \tag{A.11}$$

$$\|\bar{\mathbf{V}}_1\| = \sqrt{1+\alpha^2} \tag{A.12}$$

$$\|\bar{\mathbf{V}}_2\| = \frac{1}{\sqrt{1+\alpha^2}} \tag{A.13}$$

A.2 Proofs

A.2.1 Orthogonality of left vectors (property (A.4))

The property (A.4) results directly from the third step consisting in orthogonalizing space functions.

A.2.2 Compression (property (A.5))

The notion of compression refers to the redundancy occurring in right vectors (orthogonality implies no redundancy whereas colinearity implies a high redundancy). According to property property (A.5) to demonstrate, F is expected to compress right vectors (*i.e.* make them less colinear). Before providing its proof, we gives following elements:

- $\tilde{\boldsymbol{\phi}}_1^T \tilde{\boldsymbol{\phi}}_2 = \|\bar{\mathbf{V}}_1\| \|\bar{\mathbf{V}}_2\| \bar{\boldsymbol{\phi}}_1^T \bar{\boldsymbol{\phi}}_2 = \bar{\boldsymbol{\phi}}_1^T \bar{\boldsymbol{\phi}}_2$
- Coefficient α defined in (A.7) is assumed to be positive which can always be prescribed by choosing an appropriate sign for $\boldsymbol{\phi}_1$ or $\boldsymbol{\phi}_2$ doing for instance $\mathbf{V}_1(-\boldsymbol{\phi}_1^T) = (-\mathbf{V}_1)\boldsymbol{\phi}_1^T$. Using the Cauchy-Schwarz inequality and properties of right vectors in

(A.1) yields to:

$$0 \leqslant \boldsymbol{\phi}_1^T \boldsymbol{\phi}_2 \leqslant \sqrt{\boldsymbol{\phi}_1^T \boldsymbol{\phi}_1}\sqrt{\boldsymbol{\phi}_2^T \boldsymbol{\phi}_2} \leqslant \sqrt{\boldsymbol{\phi}_1^T \boldsymbol{\phi}_1}\sqrt{\boldsymbol{\phi}_1^T \boldsymbol{\phi}_1} \;\Rightarrow\; 0 \leqslant \boldsymbol{\phi}_1^T \boldsymbol{\phi}_2 \leqslant \boldsymbol{\phi}_1^T \boldsymbol{\phi}_1 \Rightarrow\; 0 \leqslant \alpha \leqslant 1 \quad \text{(A.14)}$$

The following quantity is developed:

$$\bar{\boldsymbol{\phi}}_1^T \bar{\boldsymbol{\phi}}_2 = \frac{1-\alpha^2}{1+\alpha^2}\boldsymbol{\phi}_1^T \boldsymbol{\phi}_2 - \frac{\alpha}{1+\alpha^2}\left[\boldsymbol{\phi}_1^T \boldsymbol{\phi}_1 - \boldsymbol{\phi}_2^T \boldsymbol{\phi}_2\right] = \boldsymbol{\phi}_1^T \boldsymbol{\phi}_2 \underbrace{\frac{(\boldsymbol{\phi}_1^T \boldsymbol{\phi}_1)(\boldsymbol{\phi}_2^T \boldsymbol{\phi}_2) - (\boldsymbol{\phi}_1^T \boldsymbol{\phi}_2)^2}{(\boldsymbol{\phi}_1^T \boldsymbol{\phi}_1)^2 + (\boldsymbol{\phi}_1^T \boldsymbol{\phi}_2)^2}}_{C_1} \quad \text{(A.15)}$$

Using the Cauchy-Schwarz inequality and properties of right vectors in (A.1) yields to:

$$0 \leqslant (\boldsymbol{\phi}_1^T \boldsymbol{\phi}_2)^2 \leqslant (\boldsymbol{\phi}_1^T \boldsymbol{\phi}_1)(\boldsymbol{\phi}_2^T \boldsymbol{\phi}_2) \;\Leftrightarrow\; 0 \leqslant (\boldsymbol{\phi}_1^T \boldsymbol{\phi}_1)(\boldsymbol{\phi}_2^T \boldsymbol{\phi}_2) - (\boldsymbol{\phi}_1^T \boldsymbol{\phi}_2)^2 \Leftrightarrow\; 0 \leqslant (\boldsymbol{\phi}_1^T \boldsymbol{\phi}_1)(\boldsymbol{\phi}_2^T \boldsymbol{\phi}_2) - (\boldsymbol{\phi}_1^T \boldsymbol{\phi}_2)^2 \leqslant (\boldsymbol{\phi}_1^T \boldsymbol{\phi}_1)^2 + (\boldsymbol{\phi}_1^T \boldsymbol{\phi}_2)^2 \quad \text{(A.16)}$$

Consequently, $\tilde{\boldsymbol{\phi}}_1^T \tilde{\boldsymbol{\phi}}_2 = C_1 \boldsymbol{\phi}_1^T \boldsymbol{\phi}_2$ with $C_1 \leqslant 1$ prooving (A.5).

A.2.3 Preservation of the ascending sorting (property (A.6))

The property (A.6) ensures that the sorting property according the norms of right vectors is preserved after the application of F. Before providing its proof, we give following elements:

- The amplitude ratio $\eta = \frac{\|\boldsymbol{\phi}_2\|^2}{\|\boldsymbol{\phi}_1\|^2} = \frac{\boldsymbol{\phi}_2^T \boldsymbol{\phi}_2}{\boldsymbol{\phi}_1^T \boldsymbol{\phi}_1} \leqslant 1$ thanks to property (A.1). Moreover, $\eta = \frac{\|\boldsymbol{\phi}_2\|^2}{\|\boldsymbol{\phi}_1\|^2} = \alpha^2 + \frac{\bar{\boldsymbol{\phi}}_2^T \bar{\boldsymbol{\phi}}_2}{\boldsymbol{\phi}_1^T \boldsymbol{\phi}_1}$. According to (A.14), α is bounded, so $0 \leqslant \alpha^2 \leqslant \eta \leqslant 1$.
- $\|\tilde{\boldsymbol{\phi}}_1\|^2 = \|\bar{\mathbf{V}}_1\|^2 \bar{\boldsymbol{\phi}}_1^T \bar{\boldsymbol{\phi}}_1 = \boldsymbol{\phi}_1^T \boldsymbol{\phi}_1 \frac{1+(2+\eta)\alpha^2}{1+\alpha^2}$
- $\|\tilde{\boldsymbol{\phi}}_2\|^2 = \|\bar{\mathbf{V}}_2\|^2 \bar{\boldsymbol{\phi}}_2^T \bar{\boldsymbol{\phi}}_2 = \boldsymbol{\phi}_1^T \boldsymbol{\phi}_1 \frac{\eta-\alpha^2}{1+\alpha^2}$

Consequently,

$$\|\tilde{\boldsymbol{\phi}}_1\|^2 \geqslant \|\tilde{\boldsymbol{\phi}}_2\|^2 \Leftrightarrow 1 + (2+\eta)\alpha^2 \geqslant \eta - \alpha^2 \quad \text{(A.17)}$$

which is true prooving property (A.6).

A.3 Iterated compression function F^{ξ}

Thanks to the properties (A.4) and (A.6) suited for inputs of F, function F can be composed to obtain an even more compressed low rank approximation. Thus, denonting ξ a functional power, one obtains the iterated function F^{ξ} such that:

$$F^{\xi} = \underbrace{F \circ F \circ \cdots \circ F}_{\xi \text{ times}} \quad \text{(A.18)}$$

The larger is the functional power ξ, the more compressed are right vectors. Let α_ξ be the coefficient defined in (A.7) at ξ^{th} iteration of F. In this section, we study the evolution of the sequence $(\alpha)_\xi$ versus ξ. One can compute, the recurrence relationship between α_ξ and $\alpha_{\xi+1}$ as well as η_ξ and $\eta_{\xi+1}$:

$$\text{given } (\eta_0, \alpha_0) : \begin{cases} \alpha_{\xi+1} = \dfrac{\alpha_\xi(\eta_\xi - \alpha_\xi^2)}{1 + \alpha_\xi^2(2 + \eta_\xi)} \\ \eta_{\xi+1} = \dfrac{\eta_\xi - \alpha_\xi^2}{1 + \alpha_\xi^2(2 + \eta_\xi)} = \dfrac{\alpha_{\xi+1}}{\alpha_\xi} \end{cases} \tag{A.19}$$

The above relationships are not obvious to formulate explicitly with ξ but some interesting elements marked or recalled:

- (α_ξ) is bounded such that $0 \leqslant \alpha_\xi \leqslant 1$ thanks to property (A.14).
- (α_ξ) decreases thanks to property (A.5).
- So, (α_ξ) converges to $\alpha_\infty = 0$ (*i.e.* right vectors become orthogonal).
- (η_ξ) is bounded such that $0 \leqslant \alpha_\xi^2 \leqslant \eta_\xi \leqslant 1$.
- If ξ sufficiently large, α_ξ becomes close to 0 and η_ξ stagnates to a value, say, η_∞.
- Asymptotically, the convergence rate of the process is:

$$\lim_{\xi \to +\infty} \frac{\alpha_{\xi+1} - \alpha_\infty}{\alpha_\xi - \alpha_\infty} = \eta_\infty$$

  Generally, η_∞ is expected to be such such $0 < \eta_\infty < 1$ providing a η_∞-linear convergence rate (geometric convergence). In practice, one can expect reasonably that $10^{-6} < \eta_\infty < 10^{-1}$. If $\eta_\xi = 0$ then $\alpha_\xi = 0$ and if $\alpha_\xi = 1$ then $\eta_\xi = 0$, the function is maximally compressed the next iteration. If η_∞ has been equal to 0, the process would have been superlinear.

- The litigious cases is for $\eta_\xi = 1$. If $\eta_\xi = 1$ and $\alpha_\xi = 1$, the next iteration leads to $\alpha_{\xi+1} = \eta_{\xi+1} = 0$. If $\eta_\xi = 1$ and $\alpha_\xi < 1$, the next iteration leads to $\alpha_{\xi+1} < \alpha_\xi$ yielding to $\eta_{\xi+1} < 1$ (standard case).

The first terms α_0 and η_0 depend on the nature of the low rank approximation to compress. In the following, subscripts ξ for both left and right vectors obtained after ξ application of F are dropped off for the sake of clarity. Convergence plots are given on figure A.1 for different values α_0 and η_0. For the case $\alpha_0 = 1 \Rightarrow \eta_0 = 1$, only 1 iteration is needed.

On figure A.2 are plot trajectories (η_ξ, α_ξ) starting from some different (η_0, α_0) in the plane (η, α). One can see evolution of both parameters (α coefficient and convergence rate). All curves go toward low values for α (convergence to zero) and show also that the

convergence rate η is stabilized to η_∞ after a few iterations. The convergence behavior is remarkably governed by the starting point (η_0, α_0):

- The higher is α_0, the lower becomes η and the higher is the convergence rate.
- The lower is η_0, the faster is the convergence.
- The lower is α_0 (right vectors are poorly correlated) and the higher η_0 (amplitudes of right vectors are similar), the lower is the convergence rate. This observation is perturbing because in this extend one has quasi-orthogonal right vectors and in other words, the job is quite completed. But the discrepancy between right vectors amplitudes is not enough to distinguish the most contributory rank one approximation.

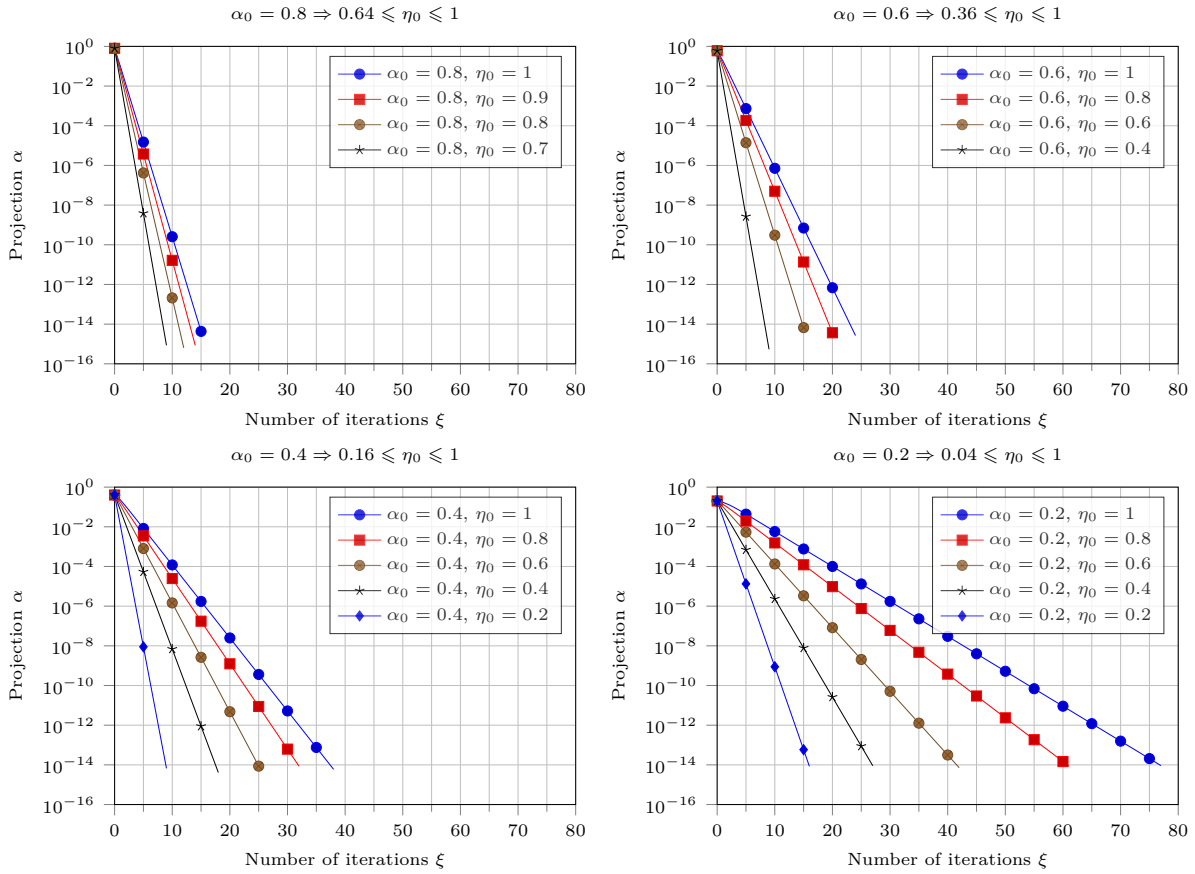


Figure A.1 – Convergence of iteration fonction F for some α_0 and η_0 characterizing some time modes.

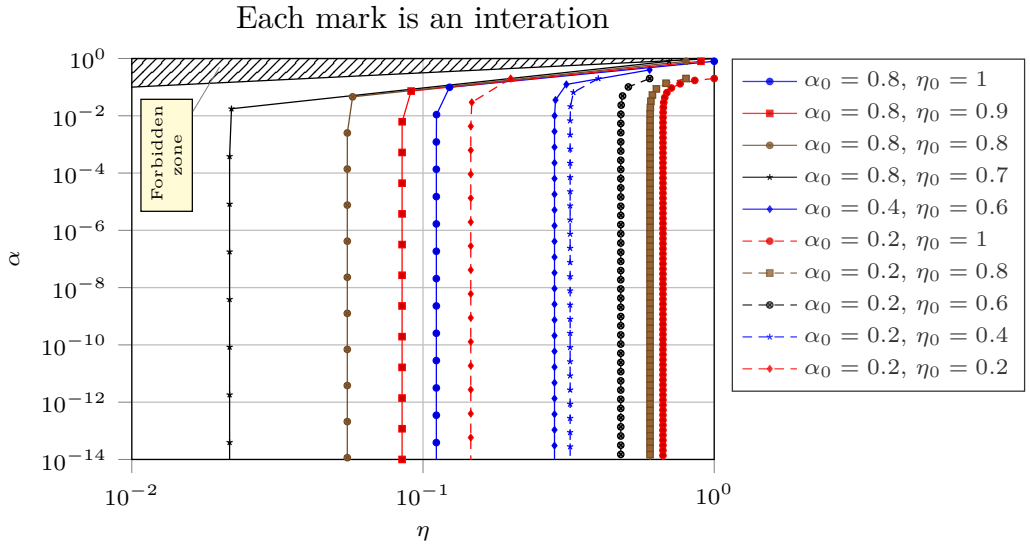


Figure A.2 – Evolution of both α_ξ and η_ξ through iteration ξ starting from some (η_0, α_0). Each mark corresponds to an iteration ξ.

Once α nullifies, both space modes ($\mathbf{V}_1$ and $\mathbf{V}_2$) and time modes ($\boldsymbol{\phi}_1$ and $\boldsymbol{\phi}_2$) are orthogonal and $\check{\mathbf{u}}$ reads:

$$\check{\mathbf{u}} = \mathbf{V}_1\boldsymbol{\phi}_1^T + \mathbf{V}_2\boldsymbol{\phi}_2^T = \begin{bmatrix}\mathbf{V}_1 & \mathbf{V}_2\end{bmatrix}\begin{bmatrix}\|\boldsymbol{\phi}_1\|_2 & \mathbf{0} \\ \mathbf{0} & \|\boldsymbol{\phi}_2\|_2\end{bmatrix}\begin{bmatrix}\frac{\boldsymbol{\phi}_1}{\|\boldsymbol{\phi}_1\|_2} & \frac{\boldsymbol{\phi}_2}{\|\boldsymbol{\phi}_2\|_2}\end{bmatrix}^T \tag{A.20}$$

Such a factorization is unique and is called the singular value decomposition of $\check{\mathbf{u}}$.

A.4 Generalization to a SVD algorithm

In this section, the compression stage is generalized with the following larger low rank approximation:

$$\check{\mathbf{u}} = \sum_{k=1}^{p}\mathbf{V}_k\boldsymbol{\phi}_k^T \quad \text{with} \quad \begin{cases}\forall(i,j)\in[\![1,p]\!]^2 : i \leqslant j \\ \mathbf{V}_i^T\mathbf{V}_j = \boldsymbol{\delta}_{ij} \\ \|\boldsymbol{\phi}_i\| \geqslant \|\boldsymbol{\phi}_j\| > 0\end{cases} \tag{A.21}$$

We introduce the following symmetric matrix $\boldsymbol{\Phi}$:

$$\boldsymbol{\Phi} = \begin{bmatrix}\boldsymbol{\phi}_1 & \boldsymbol{\phi}_2 & \cdots & \boldsymbol{\phi}_p\end{bmatrix}^T\begin{bmatrix}\boldsymbol{\phi}_1 & \boldsymbol{\phi}_2 & \cdots & \boldsymbol{\phi}_p\end{bmatrix} = \begin{bmatrix}\boldsymbol{\phi}_1^T\boldsymbol{\phi}_1 & \boldsymbol{\phi}_1^T\boldsymbol{\phi}_2 & \cdots & \boldsymbol{\phi}_1^T\boldsymbol{\phi}_p \\ \boldsymbol{\phi}_2^T\boldsymbol{\phi}_1 & \boldsymbol{\phi}_2^T\boldsymbol{\phi}_2 & \cdots & \boldsymbol{\phi}_2^T\boldsymbol{\phi}_p \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{\phi}_p^T\boldsymbol{\phi}_1 & \boldsymbol{\phi}_p^T\boldsymbol{\phi}_2 & \cdots & \boldsymbol{\phi}_p^T\boldsymbol{\phi}_p\end{bmatrix} \tag{A.22}$$

If $\boldsymbol{\Phi}$ is diagonal, then right vectors $\boldsymbol{\phi}_i$ are orthogonal. In this case and as right vectors $\mathbf{V}_i$ are orthonormal, $\check{\mathbf{u}}$ is written under its (unique) SVD.

Given ξ compression stages, all coefficients $(\alpha_\xi)_{ij} = \dfrac{\boldsymbol{\phi}_i^T \boldsymbol{\phi}_j}{\|\boldsymbol{\phi}_i\|}$ are cast into the matrix $\mathbf{A}_\xi$:

$$\mathbf{A}_\xi = \begin{bmatrix} 0 & (\alpha_\xi)_{12} & \cdots & (\alpha_\xi)_{1p} \\ 0 & 0 & \cdots & (\alpha_\xi)_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix} \tag{A.23}$$

Obviously, if $\mathbf{A}_\xi = \mathbf{0}$ then $\boldsymbol{\Phi}$ is diagonal. We propose to apply the iterated compression function F^ξ to nullify the coefficients $(\alpha_\xi)_{ij}$. For that purpose, the algorithm 17 is used.

By generalizing the previously presented proof $\forall (i,j) \in [\![1,p]\!]^2 : i \leqslant j$ (*i.e.* compression function F is applied pair-by-pair $\mathbf{V}_i, \boldsymbol{\phi}_i$ and $\mathbf{V}_j, \boldsymbol{\phi}_j$ as above), one is able to show that all coefficient $(\alpha_\xi)_{ij}$ decreases (*i.e.* $\|\mathbf{A}_{\xi+1}\|_F \leqslant \|\mathbf{A}_\xi\|_F$). Moreover, useful properties (A.1) are conserved through iteration ξ. As a consequence, algorithm 17 converges iterates until having all $(\alpha_\xi)_{ij} = 0$ (*i.e.* having the SVD decomposition for $\breve{\mathbf{u}}$).

Algorithm 17: Recursive SVD

```
while ξ ⩽ ξ_max do
    for i = p down to 2 do
        Sort the low rank approximation such that ‖φ_1‖_2 ⩾ ··· ⩾ ‖φ_p‖_2
        for j = 1 up to i − 1 do
            (V_i, φ_i, V_j, φ_j) = F(V_i, φ_i, V_j, φ_j)
        end
    end
    ξ ← ξ + 1
end
```

FOLIO ADMINISTRATIF
THÈSE SOUTENUE DEVANT L'INSTITUT NATIONAL DES SCIENCES APPLIQUÉES DE LYON

NOM : Giacoma DATE DE SOUTENANCE : 2 octobre 2014

PRÉNOM : Anthony

TITRE : Efficient acceleration techniques for nonlinear analysis of structures with frictional contact

NATURE : Doctorat NUMÉRO D'ORDRE : 2014-ISAL-0095

ÉCOLE DOCTORALE : MEGA

SPÉCIALITÉ : Mécanique - Génie Mécanique - Génie Civil

RÉSUMÉ :
La mécanique computationnelle est un outil incontournable pour le monde de l'ingénierie mécanique. Motivé par un désir de réalisme et soumis à un perpétuel gigantisme, les modèles numériques doivent aujourd'hui inclure des phénomènes physiques de plus en plus complexes. Par conséquence, d'importantes capacités calculatoires sont requises afin de traiter des problèmes à la fois non-linéaires mais aussi de grande taille. Pour atteindre cet objectif, il convient de développer les stations de calculs mais aussi les méthodes algorithmiques utilisées afin de résoudre efficacement ces types de problèmes. Récemment, les méthodes de réduction de modèle se révèlent comme d'excellentes options au développement d'algorithmes de résolution performants.
Le problème du contact frottant entre solides élastiques est particulièrement bien connu pour sa complexité et dont les temps de calcul peuvent devenir prohibitifs. En effet, les lois qui le régissent sont très hautement non-linéaires (non différentiables). Dans ce mémoire, nous nous proposons d'appliquer différentes méthodes de réduction de modèle (a posteriori et a priori) à ce type de problème afin de développer des méthodes de calculs accélérées dans le cadre de la méthode des élements finis.
Tout d'abord, en se plaçant dans le cadre des petites perturbations en évolution quasi-statique, la réductibilité de diverses solutions impliquant du contact frottant est mise en évidence via leur décomposition en valeur singulière. De plus, leur contenu à échelle séparée est exhibé. La méthode non-incrémentale et non-linéaire à large incrément de temps (LATIN) est par la suite présentée. Dans un second temps et à partir des observations faites précédemment, une méthode LATIN accélérée est proposée en s'inspirant des méthodes multigrilles non-linéaires de type "full approximation scheme" (FAS). Cette méthode s'apparente en partie aux méthodes de réduction de modèle de type a posteriori. De plus, une stratégie de calcul de modes à partir d'un modèle de substitution est proposée. Par la suite, la décomposition propre généralisée (PGD) est utilisée afin de développer une méthode de résolution non-linéaire efficace reposant fondamentalement sur une approche de réduction de modèle de type a priori. Enfin, quelques extensions sont proposées telle que la résolution de problème faisant intervenir des études paramétriques, ou encore la prise en charge de non-linéarités supplémentaires telle que la plasticité.

MOTS-CLÉS: contact frottant, LATIN, réduction de modèle, PGD, modèles de substitution.

LABORATOIRE DE RECHERCHE : LaMCoS – Laboratoire de Mécanique des Contacts et des Structures
UMR CNRS 5259 – INSA de Lyon
18-20 rue des Sciences
69621 Villeurbanne Cedex FRANCE

DIRECTEURS DE THÈSE : Pr. Gravouil Anthony et Pr. Dureisseix David

PRÉSIDENT DU JURY :

COMPOSITION DU JURY :

Pr. Pierre-Alain Boucard	Pr. Francisco Chinesta
Pr. Patrick Le Tallec	Pr. Pierre Alart
Pr. Patrick Hild	Dr. Rochette Michel
Pr. Gravouil Anthony	Pr. Dureisseix David