N°d'ordre 2013-ISAL–XXXX Année 2013

THÈSE

Réduction de modèle a priori par séparation de variables espace-temps – Application en dynamique transitoire

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

Lucas BOUCINHA
Ingénieur INSA-Lyon

Thèse soutenue le 15 novembre 2013 devant la Commission d'examen

Jury

P. LADEVÈZE	Professeur (ENS de Cachan)	Examinateur
A. HUERTA	Professeur (Universitat Politècnica de Catalunya)	Rapporteur
D. RYCKELYNCK	Maître de recherche (Mines ParisTech)	Rapporteur
F. CHINESTA	Professeur (École Centrale de Nantes)	Examinateur
A. AMMAR	Professeur (ENSAM de Angers)	Examinateur
A. GRAVOUIL	Professeur (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** 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** Secrétariat : 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 65 55 Fax : 04 78 43 37 17 Gerard.scorletti@ec-lyon.fr
E2M2	**EVOLUTION, ECOSYSTEME, MICROBIOLOGIE, MODELISATION** **http://e2m2.universite-lyon.fr** 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 : Samia VUILLERMOZ Insa : M. LAGARDE	M. Didier REVEL Hôpital Louis Pradel Bâtiment Central 28 Avenue Doyen Lépine 69677 BRON Tél : 04.72.68.49.09 Fax :04 72 68 49 16 Didier.revel@creatis.uni-lyon1.fr
INFOMATHS	**INFORMATIQUE ET MATHEMATIQUES** **http://infomaths.univ-lyon1.fr** Sec :Renée EL MELHEM	Mme Sylvie CALABRETTO Université Claude Bernard Lyon 1 INFOMATHS Bâtiment Braconnier 43 bd du 11 novembre 1918 69622 VILLEURBANNE Cedex Tél : 04.72. 44.82.94 Fax 04 72 43 16 87 infomaths@univ-lyon1.fr
Matériaux	**MATERIAUX DE LYON** **http://ed34.universite-lyon.fr** Secrétariat : 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.ec-lyon.fr** Secrétariat : 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	M. OBADIA Lionel Université Lyon 2 86 rue Pasteur 69365 LYON Cedex 07 Tél : 04.78.77.23.86 Fax : 04.37.28.04.48 Lionel.Obadia@univ-lyon2.fr

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

Résumé

La simulation numérique des phénomènes physiques est devenue un élément incontournable dans la boite à outils de l'ingénieur mécanicien. Des outils robustes et modulables, basés sur les méthodes classiques d'approximation, sont désormais couramment utilisés dans l'industrie. Cependant, ces outils nécessitent des moyens de calculs importants lorsqu'ils sont utilisés pour résoudre des problèmes complexes. Même si les progrès remarquables de l'industrie informatique rendent de tels moyens de calcul toujours plus abordables, il s'avère aujourd'hui nécessaire de proposer des méthodes d'approximation innovantes permettant de mieux exploiter les ressources informatiques disponibles. Les méthodes de réduction de modèle sont présentées comme un candidat idéal pour atteindre cet objectif. Parmi celles-ci, les méthodes basées sur la construction d'une approximation à variables séparées se sont révélées être très efficaces pour approcher la solution d'une grande variété de problèmes, réduisant les coûts numériques de plusieurs ordres de grandeur. Néanmoins, l'efficacité de ces méthodes dépend considérablement du problème traité. Dans ce manuscrit, on se propose d'évaluer l'intérêt d'une approximation à variables séparées espace-temps dans le cadre de problèmes académiques de dynamique transitoire.

On définit tout d'abord la meilleure approximation (au sens d'un problème de minimisation) de la solution d'un problème transitoire, sous la forme d'une représentation à variables séparées espace-temps. Le calcul de cette approximation étant basé sur l'hypothèse que la solution du problème de référence est connue (méthode *a posteriori*), la suite du manuscrit est dédiée à la construction d'une telle approximation sans autres connaissances a priori sur la solution de référence, que les opérateurs du problème espace-temps dont elle est solution (méthode *a priori*). Un formalisme générique, basé sur une représentation tensorielle des opérateurs du problème espace-temps est alors introduit dans un cadre multichamps. On développe ensuite un solveur exploitant ce format générique, pour construire une approximation à variables séparées espace-temps de la solution d'un problème transitoire. Ce solveur est basé sur la décomposition généralisée propre de la solution (Proper Generalized Decomposition - PGD). Un état de l'art des algorithmes existants permet alors d'évaluer l'efficacité des définitions classiques de la PGD pour approcher la solution de problèmes académiques de dynamique transitoire. Les résultats obtenus mettant en défaut l'optimalité de la PGD la plus robuste, une nouvelle définition, récemment introduite dans la littérature, est appliquée dans un cadre multichamps à la résolution d'un problème d'élastodynamique 2D. Cette nouvelle définition, basée sur la minimisation du résidu dans une norme idéale, permet finalement d'obtenir une très bonne approximation de la meilleure approximation de rang donné, sans avoir à calculer un grand nombre de modes espace-temps.

MOTS CLÉS: dynamique transitoire, séparation de variables espace-temps, POD/PGD

Table des matières

Introduction

Des outils robustes mais nécessitant des moyens de calculs importants

Les outils de simulation sont aujourd'hui suffisamment robustes pour être exploités dans un contexte industriel. Ils sont devenus indispensables pour valider, voir certifier la conception des produits, ou bien permettre leur optimisation. Les phénomènes physiques modélisés sont de plus en plus complexes. Cependant, la complexité des modèles qu'il est possible de développer à l'aide des outils de simulation est limitée par les moyens de calcul dont on dispose. On est ainsi rapidement confronté à deux problèmes : la durée de la simulation et l'espace mémoire nécessaire pour effectuer le calcul et stocker les résultats.

Un exemple est la simulation du comportement dynamique d'une structure soumise à un choc. Pour ce type de problème, la méthode des éléments finis et les schémas d'intégration en temps sont des outils très robustes. Ils requièrent cependant des discrétisations spatiale et temporelle très fines, pour représenter les variations locales du champ de déplacement avec une précision raisonnable. Supposons que l'on ait discrétisé le champ de déplacement de la structure avec n_S dégrés de liberté et que la simulation de son évolution au cours du temps nécessite n_T pas de temps. On est alors ammené à résoudre des systèmes linéaires de très grande taille ($n_S \times n_S$), un très grand nombre de fois (n_T fois). Et le temps nécessaire pour effectuer la simulation devient trop long dès lors que les dimensions n_S et n_T sont importantes. Le stockage de la solution sur le domaine espace-temps est tout aussi problématique (on doit stocker $n_S n_T$ valeurs).

Les progrès remarquables réalisés dans l'industrie informatique ont permis de traiter des problèmes toujours plus grands avec les méthodes classiques d'approximation. Cependant, il s'avère aujourd'hui nécessaire de proposer des méthodes d'approximation innovantes permettant de mieux exploiter les ressources informatiques disponibles.

Une rupture méthodologique : la réduction de modèle par séparation de variables

Les méthodes de réduction de modèle ont été introduites pour atteindre cet objectif. Parmis celles-ci, les méthodes basées sur la construction d'une approximation à variables séparées se sont révélées être très efficaces pour approcher

la solution d'une grande variété de problèmes [Ladevèze, 1999, Ammar *et al.*, 2006, Nouy, 2007, Chinesta *et al.*, 2011]. Dans certains cas, elles ont permis de réduire les coûts numériques de plusieurs ordres de grandeur. Néanmoins, l'efficacité de ces méthodes dépend considérablement du problème traité. Aussi, l'objectif poursuivi de ce manuscrit est d'évaluer l'intérêt d'une approximation à variables séparées espace-temps dans le cadre de problèmes académiques de dynamique transitoire.

La stratégie proposée repose sur deux points clés :

- Le premier point concerne la représentation d'un champ défini sur le domaine espace-temps, sous la forme d'une somme de produits de fonctions définies sur l'espace et le temps, chaque produit de fonctions pouvant être interprété comme un mode espace-temps. De cette manière, on remplace le stockage de $n_S n_T$ valeurs par le stockage de $M(n_S + n_T)$ valeurs, où M est le nombre de modes espace-temps utilisés. Une telle représentation à variables séparées permettra donc de réduire considérablement l'espace mémoire nécessaire au stockage d'un champ sur le domaine espace-temps dès lors que le nombre de modes espace-temps M sera très faible par rapport aux dimensions n_S et n_T. L'enjeu est alors de savoir quel est la valeur de M qui permette d'approcher précisemment un champ connu, sous la forme d'une représentation à variables séparées.
- Le second point concerne le développement d'un solveur non-incrémental permettant d'approcher la solution (non connue) d'un problème transitoire sous la forme d'une représentation à variables séparées espace-temps. Le solveur développé dans ce manuscrit est basé sur la décomposition généralisée propre, mieux connue sous son acronyme anglais PGD (« Proper Generalized Decomposition »). L'idée est d'exploiter le caractère séparable des opérateurs d'un problème transitoire, de façon à remplacer la résolution du problème sur le domaine espace-temps par une succession de résolutions alternatives d'un problème spatial et d'un problème temporel. On remplace ainsi la résolution de n_T systèmes linéaires de taille $n_S \times n_S$ par $M\xi$ résolutions d'un système linéaire de taille $n_S \times n_S$ et d'un autre de taille $n_T \times n_T$, où ξ est un nombre d'itérations. Dans le cas où $n = n_S = n_T$, un tel solveur permettra donc de réduire le temps de calcul si $M\xi << n$.

L'objectif général poursuivi dans ce manuscrit est d'évaluer l'efficacité de ces deux points clés dans le cas de problèmes académiques de dynamique transitoire.

Organisation du manuscrit

Le manuscrit est organisé autour de la réalisation de cet objectif.

- Dans le Chapitre 1, on présente tout d'abord les techniques d'approximation classiquement utilisées pour approcher la solution d'un problème de dynamique transitoire. Le problème de référence est introduit, puis les techniques d'approximation spatiale et temporelle sont détaillées et leurs propriétés numériques sont

commentées.

- Le Chapitre 2 est consacré à l'évaluation du premier point clé de la stratégie. On définit la meilleure approximation d'un champ (connu) sous la forme d'une représentation à variables séparées espace-temps, au sens d'un problème de minimisation. Les outils permettant de construire cette meilleure approximation sont rapidement présentés, puis on évalue l'efficacité d'une telle approximation en terme de gain mémoire, dans le cas de problème unidimensionnel de dynamique transitoire.

La construction de l'approximation à variables séparées présentée dans le Chapitre 2 est appelée la décomposition *a posteriori* d'un champ, car celui-ci est calculé (et donc également stocké) dans une étape préliminaire avant d'être approché par une représentation à variables séparées. Aussi, un problème plus difficile est de construire une approximation à variables séparées de ce champ, sans autres connaissances *a priori* sur celui-ci que les opérateurs du problème espace-temps dont il est solution. On verra qu'un problème encore plus difficile est de trouver *a priori* une bonne approximation de la meilleure approximation à variables séparées introduite au Chapitre 2. Ces aspects sont l'objet du second point de la stratégie proposée et les chapitres suivants y sont dédiés.

- Le Chapitre 3 est une parenthèse dans la présentation de la stratégie suivie. On évalue l'efficacité des méthodes de réduction de modèle les plus simples et les plus populaires, classiquement utilisées en dynamique des structures. Ces méthodes sont basées sur la projection du problème de référence sur une base de fonctions spatiales de dimension réduite par rapport à la dimension de l'espace d'approximation. Elles aboutissent également à une approximation à variables séparées espace-temps et peuvent donc être comparées à la meilleure approximation définie au Chapitre 2. Très efficaces dans le cadre de problèmes basses fréquences, ces méthodes de réduction de modèle le sont beaucoup moins pour approcher la solution d'un problème de choc, justifiant ainsi le développement de nouvelles stratégies.
- Le Chapitre 4 est le plus technique du manuscrit. On introduit un formalisme général pour représenter les opérateurs d'un problème espace-temps sous la forme d'une somme de produits tensoriels d'opérateurs spatiaux et temporels. Ce formalisme est étendu dans le cas d'un problème à F-champs. La construction des opérateurs du problème dans ce format est illustrée pour les différentes méthodes d'approximation introduites au Chapitre 1 (méthodes éléments finis en espace et en temps, schéma d'intégration en temps). Une attention particulière est apportée à la prise en compte des conditions aux limites et initiales dans le cadre d'une telle représentation.

La représentation tensorielle des opérateurs introduite au Chapitre 4 peut ensuite être exploitée pour développer des solveurs génériques, capables de construire une

approximation à variables séparées de la solution d'un problème donné sous format tensoriel. Le développement de tels solveurs est l'objet des deux derniers chapitres.

- Le Chapitre **??** est un état de l'art des algorithmes utilisés dans le cadre de la méthode de décomposition généralisée propre (PGD). Les définitions classiques de la PGD, ainsi que les algorithmes de construction associés, sont décrits dans le cas d'un problème espace-temps à un champ, puis dans le cas d'un problème multichamps. L'efficacité de l'ensemble des algorithmes présentés est alors évaluée dans le cas de différents problèmes académiques de dynamique transitoire.
- Les résultats obtenus dans le Chapitre **??** mettant en défaut l'optimalité de la PGD la plus robuste, le Chapitre 6 est consacré à la présentation d'une nouvelle définition, récemment introduite dans la littérature [Billaud-Friess *et al.*, 2013]. Cette nouvelle définition est appliquée dans un cadre multichamps à la résolution d'un problème d'élastodynamique bidimensionnel. Basée sur la minimisation du résidu dans une norme idéale, elle permet finalement d'obtenir une très bonne approximation de la meilleure approximation définie au Chapitre 2, sans avoir à calculer plus de modes espace-temps que nécessaires.

Des conclusions et perspectives à ces travaux sont finalement proposées.

Les notations et le point de vue adopté pour la définition des espaces produits tensoriels sont précisés dans l'Annexe A.

Chapitre 1

Méthodes classiques d'approximation en dynamique transitoire

Ce chapitre présente les méthodes d'approximation spatiale et temporelle, classiquement utilisées pour approcher la solution d'un problème de dynamique transitoire.

Sommaire

1.1 Problème de référence

Afin de simplifier la présentation, on décrit dans ce chapitre la modélisation du problème d'élastodynamique dans un milieu unidimensionnel. La modélisation dans un milieu bidimensionnel est présentée dans le Chapitre 4.

1.1.1 Formulation forte

On cherche à prédire la réponse dynamique d'une poutre occupant le domaine spatial $\Omega = [0, L]$, au cours de l'intervalle de temps $I = [0, T]$. On considère la réponse de la poutre en traction-compression. Celle-ci est décrite par les champs scalaires de déplacement longitudinal, noté $u(x,t)$ et de contrainte de traction, noté $\sigma(x,t)$ en tout point (x,t) de $\Omega \times I$. La poutre est soumise, au cours de l'intervalle de temps I, à un déplacement $g(t)$ imposé sur son extrémité $\partial\Omega_u$ et à un effort ponctuel $p(t)$ imposé son extrémité $\partial\Omega_\sigma$, avec la partition suivante $\partial\Omega = \partial\Omega_u \cup \partial\Omega_\sigma$ et $\partial\Omega_u \cap \partial\Omega_\sigma = \varnothing$. L'état à l'instant initial est connu et décrit par les champs de déplacement $u_0(x)$ et de vitesse $v_0(x)$, définis sur Ω. Le matériau est caractérisé par sa densité ρ et son module d'élasticité E. La section de la poutre est notée A. Dans ce cadre, la réponse de la poutre est gouvernée par les équations suivantes qui constituent le problème de référence :

Problème 1.1. Le problème de référence consiste à trouver les champ de déplacement $u(x,t)$ et de contrainte $\sigma(x,t)$ suffisamment réguliés, qui vérifient :

- les équations de liaisons et conditions initiales,

$$u = g\,, \qquad \forall (x,t) \in \partial\Omega_u \times I\,, \tag{1.1a}$$

$$u = u_0\,, \qquad \forall (x,t) \in \Omega \times \{0\}\,, \tag{1.1b}$$

$$\frac{\partial u}{\partial t} = v_0\,, \qquad \forall (x,t) \in \Omega \times \{0\}\,, \tag{1.1c}$$

- les équations d'équilibre,

$$\frac{\partial \sigma}{\partial x} = \rho \frac{\partial^2 u}{\partial t^2}\,, \qquad \forall (x,t) \in \Omega \times I\,, \tag{1.1d}$$

$$\sigma = \frac{p}{A}\,, \qquad \forall (x,t) \in \partial\Omega_\sigma \times I\,, \tag{1.1e}$$

- et la relation de comportement,

$$\sigma = E\epsilon\,, \qquad \forall (x,t) \in \Omega \times I\,, \tag{1.1f}$$

où le champ de déformation ϵ est donné par $\epsilon(u) = \frac{\partial u}{\partial x}$.

Remarque 1.1. *Lorsque la relation de comportement contient un terme d'amortissement, la réponse de la structure peut être décomposée en la somme d'une composante transitoire et d'une composante stabilisée. Du fait de l'amortissement, la composante transitoire s'annulle au bout d'un certain temps. La réponse stabilisée de la structure est alors essentiellement due aux sollicitations extérieures. Dans le cadre de ces travaux, on s'intéresse à la réponse transitoire de la structure (la durée de la simulation est telle que l'on peut considérer un comportement purement élastique (non-amorti) du matériau). Dans ce cas, il est naturel d'adopter une démarche de modélisation en variables espace-temps. Cependant, lorsque l'on cherche à prédire la réponse stabilisée de la structure (et donc que la loi de comportement contient un terme d'amortissement), il est peut-être avantageux de considérer une modélisation du problème en variables espace-fréquence [Ohayon et Soize, 1998].*

1.1.2 Comportement dynamique suite à un choc

On s'intéresse plus particulièrement à la réponse dynamique d'une structure soumise à un choc. Un choc est caractérisé par une sollicitation d'une durée finie, notée ΔT, rapide par rapport à une durée caractéristique de la structure. La chute d'un objet sur une structure, un choc pyrotechnique, ou encore un déplacement imposé du à un séisme sont des exemples d'une telle sollicitation. On considère ici une modélisation simplifiée et on représente une sollicitation de choc à l'aide de la fonction suivante :

$$\begin{aligned} & choc(x,t;\Delta T) = choc^S(x)\, choc^T(t;\Delta T) \\ \text{avec} \quad & choc^T(t;\Delta T) = \begin{cases} \frac{1}{2}\left(1-\cos(\frac{2\pi}{\Delta T}t)\right) & \text{si } t \in [0,\Delta T] \\ 0 & \text{si } t \notin [0,\Delta T] \end{cases}, \end{aligned} \tag{1.2}$$

où la fonction $choc^S(x)$ est la composante spatiale du choc (dont le support est en général localisé sur une petite portion de la frontière de la structure), et la fonction $choc^T(t)$ est sa composante temporelle (dont le support est localisé sur l'intervalle de temps $[0,\Delta T]$, voir la Figure 1.1).

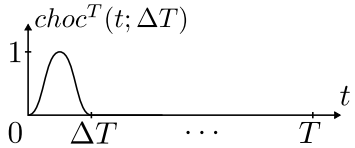

FIGURE 1.1: Représentation simplifiée de la composante temporelle d'une sollicitation de choc.

Un choc est à l'origine de la propagation d'une onde dans la structure. C'est-à-dire qu'il initie une perturbation locale du milieu qui n'est pas détectée instantanément dans le reste de la structure. Cette perturbation se propage de proche en proche à une

vitesse c qui dépend des propriétés d'élasticité et d'inertie du matériau. Une structure étant un domaine borné, la perturbation « rebondit » lorsqu'elle atteint une extrémité du domaine. On observe alors différents comportements dynamiques, selon que la durée ΔT caractérisant la sollicitation est plus ou moins longue par rapport au temps que la perturbation qu'elle provoque, met pour se propager le long d'une dimension caractéristique de la structure (notée L). On choisit, dans ce manuscrit, de caractériser la réponse de la structure suite à un choc, à l'aide du nombre adimensionné κ, défini comme suit :

$$\boxed{\kappa = \left(\frac{L}{c\Delta T}\right)^2}. \tag{1.3}$$

La structure est décrite par la durée L/c où L est une distance caractéristique entre le point d'application du choc et un point de la frontière $\partial\Omega$, et c est la célérité des ondes dans le milieu. Le nombre κ compare donc la durée du choc ΔT à la durée nécessaire à l'onde (provoquée par le choc) pour se propager d'un bout à l'autre de la structure. Une interprétation géométrique du nombre κ est donnée dans l'Exemple 1.1.

L'exposant dans la définition (1.3) provient du lien entre le nombre κ et les équations du problème de référence. En recombinant l'équation d'équilibre (1.1d) et la loi de comportement (1.1f) en privilégiant le déplacement, on montre facilement que le champ de déplacement, solution du problème de référence, vérifie l'équation des ondes suivante :

$$\frac{\partial^2 u}{\partial x^2} = \frac{1}{c^2}\frac{\partial^2 u}{\partial t^2}, \tag{1.4}$$

où $c = \sqrt{E/\rho}$ est la célérité des ondes dans le milieu. En reformulant cette équation avec les coordonnées adimensionnelles $\tilde{x} = \frac{x}{L}$ et $\tilde{t} = \frac{t}{\Delta T}$, on obtient l'équation suivante, faisant intervenir le nombre adimensionné κ :

$$\frac{\partial^2 u}{\partial \tilde{x}^2} = \kappa \frac{\partial^2 u}{\partial \tilde{t}^2}. \tag{1.5}$$

Ainsi lorsque $\kappa << 1$, le terme d'accélération dans l'équation (1.5) est négligeable devant le terme laplacien. Dans ce cas, la réponse de la structure est quasi-statique : le déplacement est nul à partir du moment où la structure n'est plus sollicitée et une modélisation dynamique n'a alors pas d'intérêt. Par contre, si le nombre κ est du même ordre de grandeur ou supérieure à l'unité, alors le terme d'accélération devient prépondérant par rapport au terme laplacien et une modélisation dynamique est incontournable. Cet aspect est illustré dans l'Exemple 1.1.

Remarque 1.2. *Dans le cas d'un milieu bi- ou tri-dimensionnel, le champ de déplacement, solution du problème d'élastodynamique, peut être représenté par l'intermédiaire*

des potentiels de Lamé, Ψ_L et $\mathbf{\Psi}_T$, vérifiant les équations des ondes suivantes :

$$\Delta(\Psi_L) = \frac{1}{c_L^2}\frac{\partial^2 \Psi_L}{\partial t^2} \quad avec \quad c_L^2 = \frac{\lambda + 2\mu}{\rho}, \tag{1.6a}$$

$$et \quad \Delta(\mathbf{\Psi}_T) = \frac{1}{c_T^2}\frac{\partial^2 \mathbf{\Psi}_T}{\partial t^2} \quad avec \quad c_T^2 = \frac{\mu}{\rho}. \tag{1.6b}$$

Ces potentiels caractérisent différents modes de propagation des ondes dans le milieu [Achenbach, 1973]. Un champ de déplacement de la forme $\nabla(\Psi_L)$ décrit ainsi des ondes longitudinales se propageant à la célérité c_L alors qu'un champ de déplacement de la forme rot$(\mathbf{\Psi}_T)$ décrit des ondes transversales se propageant à la célérité c_T. On peut alors introduire comme pour le cas unidimensionnel, les nombres sans dimension $\kappa_L = (\frac{L}{c_L \Delta T})^2$ et $\kappa_T = (\frac{L}{c_T \Delta T})^2$, caractérisant le comportement dynamique de la structure en réponse à un choc.

Exemple 1.1. (Régimes basse et moyenne fréquences) Dans cet exemple, on illustre le comportement dynamique d'une poutre en réponse à un choc pour différentes valeurs du nombre sans dimension κ. Le problème considéré est représenté sur la Figure 1.2. Le déplacement est imposé nul au point $x = L$ et les conditions initiales sont nulles. Un déplacement $g(t;\Delta T)$ de la forme de (1.2), d'une durée caractéristique ΔT, est appliqué au point $x = 0$. Dans ce cas, le nombre $\kappa = \left(\frac{L}{c\Delta T}\right)^2$ compare la durée de la sollicitation ΔT et le temps $\frac{L}{c}$ mis par l'onde provoquée par le choc pour atteindre l'autre extrémité de la poutre. Plus le nombre κ est grand et plus l'aire colorée en bleu sur la Figure 1.2 est réduite par rapport au reste du domaine espace-temps.

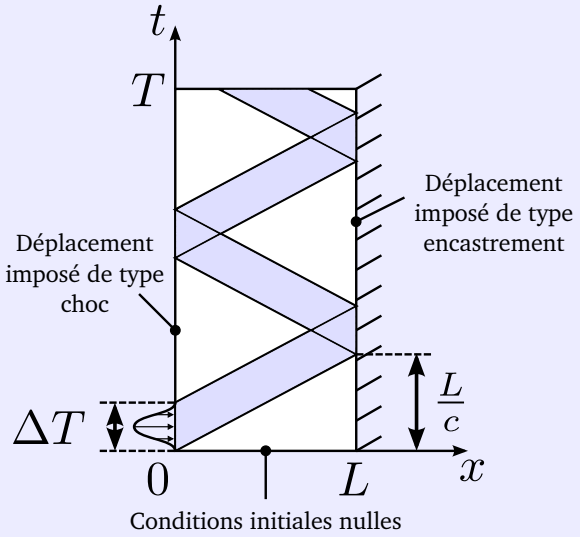

FIGURE 1.2: Description espace-temps de la propagation d'ondes dans un milieu 1D. La zone bleutée correspond aux portions du domaine espace-temps où le déplacement est non-nul.

La modélisation du problème consiste à trouver le déplacement $u(x,t)$ en tout point $(x,t) \in [0,L] \times [0,T]$ qui vérifie :

$$\frac{\partial^2 u}{\partial x^2} = \frac{1}{c^2}\frac{\partial^2 u}{\partial t^2} \quad \text{avec} \quad \begin{cases} u(0,t) & = g(t;\Delta T) \\ u(L,t) & = 0 \\ u(x,0) & = 0 \\ \partial u/\partial t(x,0) & = 0 \end{cases}. \tag{1.7}$$

La solution exacte de ce problème peut être obtenue analytiquement en appliquant la transformée de Laplace en temps à l'équation d'équilibre [Géradin et Rixen, 1997]. Après application de la transformée inverse, on obtient la solution du problème sous la forme d'une somme finie de termes :

$$u(x,t) = \sum_{n=0}^{n^+} g\left(t - \frac{x+2Ln}{c};\Delta T\right) - \sum_{n=0}^{n^-} g\left(t + \frac{x-2L(n+1)}{c};\Delta T\right), \tag{1.8}$$

où les entiers n^+ et n^- sont bornés par $0 \leq n^+ \leq \frac{cT-x}{2L}$ et $0 \leq n^- \leq \frac{cT+x}{2L} - 1$ pour tout $x \in [0,L]$. Pour simplifier l'analyse, on choisit $L = 1\text{m}$, $c = 1\text{m/s}$ et $T = 10\text{s}$.

Afin de caractériser le comportement dynamique de la structure, on représente la solution sur le domaine espace-temps pour les valeurs de $\kappa = 0.01 - 0.1 - 1 - 10 - 100$ (voir la Figure 1.3). Pour des valeurs de κ très faibles, on constate que la solution est très proche de la solution quasi-statique (voir la solution pour $\kappa = 0.01$ qui correspond à $\Delta T = 10$). L'amplitude du déplacement est négligeable dès lors que la sollicitation s'annule. Pour les valeurs de κ plus grandes, la réponse dynamique est totalement différente de la solution quasi-statique (voir la solution pour $\kappa = 100$ qui correspond à $\Delta T = 0.1$). À partir des valeurs de $\kappa \geq 1$, on distingue clairement la propagation de la perturbation dans le domaine espace-temps : la solution présente de forts gradients par rapport aux variables spatiale et temporelle, dont le support est localisé à différents lieux du domaine espace-temps.

Afin de qualifier le comportement oscillant de la réponse, on calcule également la transformée de Fourier discrète de la solution, par rapport à la variable temporelle, pour chaque valeur de $x \in \Omega$. On obtient alors le champ complexe $\widehat{u}(x,\omega)$ dont le module est représenté dans le domaine espace-fréquence (pour les valeurs de ω telles que $\frac{\omega}{2\pi} = \frac{n}{T}$ avec $n = 1,2,\cdots$) sur la Figure 1.3. On observe alors différents pics pour les valeurs de $\frac{\omega}{2\pi}$ correspondant aux fréquences propres de vibration de la structure (pour $\frac{\omega}{2\pi} = \frac{n}{2}$). À chaque fréquence propre est associée un mode propre de vibration de la structure. Ainsi, pour des faibles valeurs de κ, seulement les premiers modes de vibration contribuent à la réponse de la structure. On appellera ce comportement dynamique de la structure, le régime basse fréquence et on le caractérisera par les valeurs de $\kappa \in [0.01;1]$. Lorsque la valeur de κ augmente et devient grande devant l'unité, de

plus en plus de modes contribuent de façon non négligeable à la réponse. On appelera ce comportement dynamique le régime moyenne fréquence et on le caractérisera par les valeurs de $\kappa \in [1;100]$.

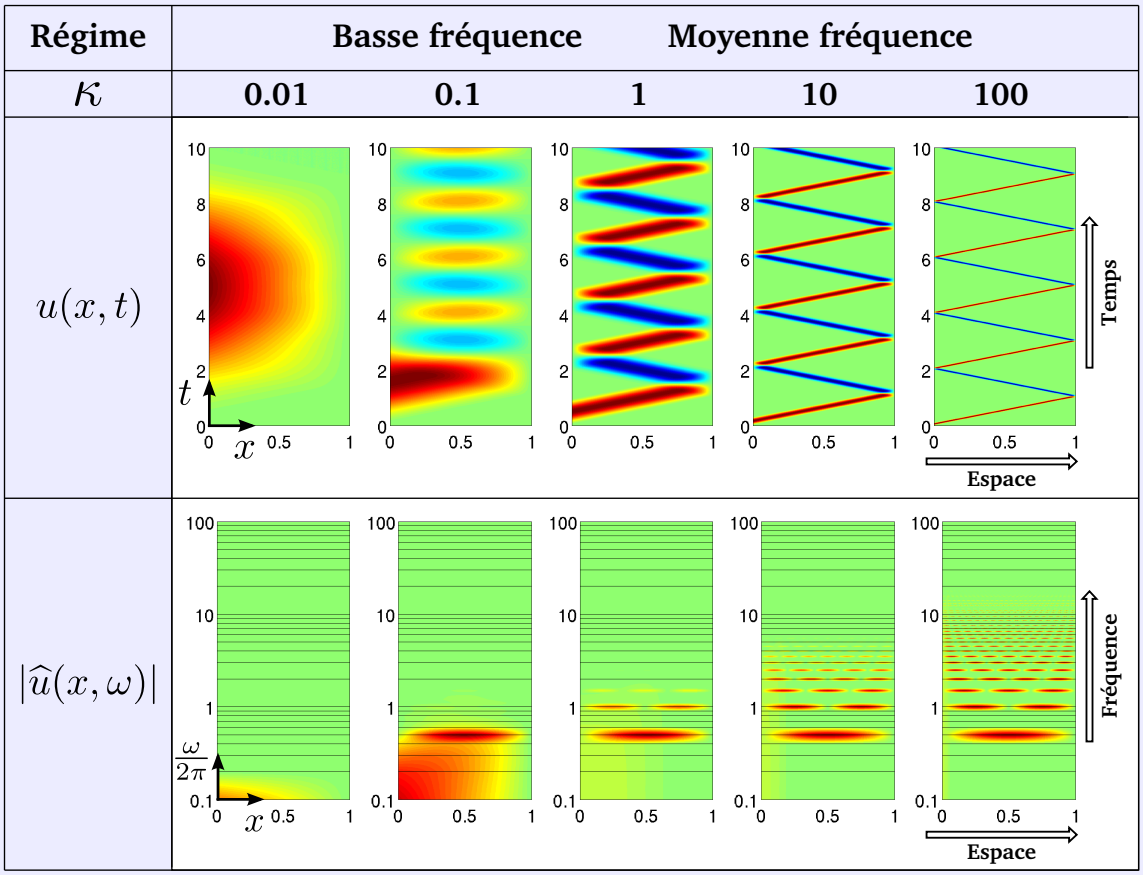

FIGURE 1.3: Comparaison des régimes dynamiques basse & moyenne fréquence pour l'équation des ondes 1D. Le déplacement $u(x,t)$ est représenté sur le domaine espace-temps et le module de sa transformée de Fourier $\widehat{u}(x,\omega)$ est représenté sur le domaine espace-fréquence, pour différentes valeurs de $\kappa = \left(\frac{L}{c\Delta T}\right)^2$.

Le Problème 1.1 ne peut être résolu analytiquement que pour des géométries et des conditions aux limites simplifiées. Dans le cas général, la solution du problème est approximée en espace et en temps à l'aide d'outils numériques. Les deux prochaines sections présentent les techniques classiques d'approximation, respectivement en espace et en temps, que l'on utilise dans la suite du manuscrit pour résoudre numériquement le problème d'élastodynamique.

1.2 Approximation du problème en espace

Dans cette section, on montre comment construire une approximation spatiale de la solution (en déplacement) du problème d'élastodynamique, par la méthode des éléments finis. On introduit pour cela une formulation faible du problème à chaque instant. On cherche alors une approximation de la solution exacte dans un espace de dimension finie, construit à partir d'une base de fonctions de type éléments finis. On aboutit ainsi au problème semi-discrétisé en espace et continue en temps.

1.2.1 Formulation faible

Les équations d'équilibre (1.1d) et (1.1e) sont définies localement. Une formulation faible du problème consiste à écrire ces équations sous la forme d'une intégrale sur le domaine Ω, à chaque instant. Cette formulation, dite globale, peut être obtenue directement à partir du principe des puissances virtuelles, ou bien en intègrant les équations d'équilibre sur Ω après les avoir multipliées par une fonction test $u^\star(x)$. Différentes stratégies peuvent alors être employées pour imposer les équations de liaison et conditions initiales. On choisit ici d'imposer les équations de liaison de façon forte : le champ inconnu $u(x,t)$ est cherché, à chaque instant, dans l'espace des fonctions définies de Ω dans $\mathbb{R}$ et satisfaisant a priori l'équation (1.1a). Cet espace, noté [1] $\mathscr{U}^S(\Omega;g)$ est appelé l'espace des fonctions cinématiquement admissibles à g et est défini par

$$\mathscr{U}^S(\Omega;g) = \left\{ u \in \mathscr{U}^S(\Omega) \mid u(x) = g(x)\,, \quad \forall x \in \partial\Omega_u \right\}, \tag{1.9}$$

où $\mathscr{U}^S(\Omega)$ est l'ensemble des fonctions définies de Ω dans $\mathbb{R}$, continues et suffisamment régulières. Le champ test $u^\star(x)$ est pris dans l'espace $\mathscr{U}^S(\Omega;0)$ des fonctions cinématiquement admissibles à zéro, permettant ainsi d'annuler la puissance virtuelle des efforts de liaison agissant sur $\partial\Omega_u$. La variable temps est ici prise comme un paramètre, c'est-à-dire que l'on identifie le champ $u(x,t)$ à une fonction définie de I dans $\mathscr{U}^S(\Omega;g(t))$ (continue et suffisamment régulières).

Problème 1.2. Une formulation faible en espace du Problème 1.1 consiste à trouver le champ de déplacement $u : I \to \mathscr{U}^S(\Omega;g(t))$ qui vérifie :

$$m(u^\star, \frac{\partial^2 u}{\partial t^2}) + k(u^\star, u) = f(u^\star;t)\,, \quad \forall u^\star \in \mathscr{U}^S(\Omega;0)\,, \tag{1.10a}$$

$$\text{avec} \quad u(x,0) = u_0\,, \quad \text{pour } x \in \Omega\,, \tag{1.10b}$$

$$\text{et} \quad \frac{\partial u}{\partial t}(x,0) = v_0\,, \quad \text{pour } x \in \Omega\,. \tag{1.10c}$$

1. L'exposant « S » dans la notation des espaces fonctionnels indique des fonctions définies sur le domaine spatial Ω. On notera $\mathscr{U}^S$ (sans mentionner les détails) lorsqu'il n'y a pas d'ambiguïté sur l'espace considéré.

où les produits scalaires $m(.,.)$ et $k(.,.)$, et la forme linéaire $f(.;t)$ sont définis comme suit :

$$m(u^\star, u) = \int_\Omega \rho\, A\, u^\star\, u \,\mathrm{d}x, \tag{1.10d}$$

$$k(u^\star, u) = \int_\Omega EA \frac{\mathrm{d}u^\star}{\mathrm{d}x} \frac{\mathrm{d}u}{\mathrm{d}x} \,\mathrm{d}x, \tag{1.10e}$$

$$\text{et} \quad f(u^\star; t) = u^\star(x)\, p(x,t) \quad \text{pour } x \in \partial\Omega_\sigma\,. \tag{1.10f}$$

Remarque 1.3. *La régularité imposée dans les espaces fonctionnelles n'a pas été précisée. Ce choix peut être relié à des considérations énergétiques : grossièrement, on impose que l'énergie libre et l'énergie cinétique du système aient des valeurs finies sur le domaine* Ω *à tout instant* $t \in I$*. On suppose dans ce manuscrit que le problème est bien posé et qu'il admet une unique solution, sans plus entrer dans les détails [Allaire, 2005].*

Remarque 1.4. *En pratique, on construit un champ* u*, cinématiquement admissible à* g*, à partir d'un champ* $\tilde{u}$ *(inconnu) cinématiquement admissible à zéro, et d'un champ* $\tilde{g}$ *(connu) défini sur* Ω *et égale à* g *sur* $\partial\Omega_u$*, c'est-à-dire :*

$$u \in \mathscr{U}^S(\Omega; g) \Leftrightarrow u = \tilde{u} + \tilde{g} \quad \textit{avec} \quad \begin{cases} \tilde{u} \in \mathscr{U}^S(\Omega; 0) \\ \tilde{g} = g \textit{ sur } \partial\Omega_u \end{cases} . \tag{1.11}$$

La formulation faible que l'on résoud en pratique, consiste donc à trouver $\tilde{u} : I \to \mathscr{U}^S(\Omega;0)$ *tel que*

$$m(u^\star, \frac{\partial^2 \tilde{u}}{\partial t^2}) + k(u^\star, \tilde{u}) = \tilde{f}(u^\star; t)\,, \quad \forall\, u^\star \in \mathscr{U}^S(\Omega;0)\,, \tag{1.12a}$$

$$\textit{avec} \quad \tilde{u}(x,0) = \tilde{u}_0\,, \quad \textit{pour } x \in \Omega\,, \tag{1.12b}$$

$$\textit{et} \quad \frac{\partial \tilde{u}}{\partial t}(x,0) = \tilde{v}_0\,, \quad \textit{pour } x \in \Omega\,, \tag{1.12c}$$

où le produit scalaire $\tilde{f}(.;t)$ *est défini par*

$$\tilde{f}(u^\star; t) = f(u^\star; t) - m(u^\star, \frac{\partial^2 \tilde{g}}{\partial t^2}) - k(u^\star, \tilde{g})\,, \tag{1.13}$$

et les champs initiaux $\tilde{u}_0$ *et* $\tilde{v}_0$ *proviennent de la décomposition des champs* u_0 *et* v_0 *sous la forme de (1.11).*

1.2.2 Semi-discrétisation

Un des intérêts d'une formulation faible est de pouvoir se ramener à la résolution d'un système d'équations linéaires. On utilise pour cela la méthode de Galerkin qui consiste à remplacer l'espace $\mathscr{U}^S$ par un sous-espace de dimension finie, noté[2] $\mathscr{U}^S_h$.

2. L'indice « h » dans la notation des espaces fonctionnels indique un espace de dimension finie n_S avec dans le cas général $h = 1/n_S$.

Cet espace d'approximation est construit de la façon suivante :

$$\mathscr{U}_h^S = \left\{ u \in \mathscr{U}^S \mid u(x) = \sum_{i=1}^{n_S} U_i \phi_i(x) \right\}, \tag{1.14}$$

où les éléments $[\phi_1, \cdots, \phi_{n_S}]$ forment une famille libre de $\mathscr{U}^S$. On obtient alors une approximation $\tilde{u}_h$ de $\tilde{u}$ en substituant $\mathscr{U}^S$ par $\mathscr{U}_h^S$ dans les équations (1.12). Cette approximation peut s'écrire sous la forme $\tilde{u}_h(x,t) = \boldsymbol{\phi}(x).\boldsymbol{U}(t)$ où $\boldsymbol{U}(t)$ est le vecteur des coordonnées du champ $\tilde{u}_h$ dans la base d'approximation $\boldsymbol{\phi} = [\phi_1, \cdots, \phi_{n_S}]$, à chaque instant. En exprimant de même, le champ test dans l'espace d'approximation et en remarquant que l'équation (1.12a) est vrai quel que soit le champ test, on aboutit à un système d'équations différentielles ordinaires du second ordre, dont l'inconnue est le vecteur $\boldsymbol{U}(t)$. Le système d'équations obtenu est appelé le problème semi-discrétisé en espace.

Problème 1.3. Le problème semi-discrétisé en espace consiste à trouver le vecteur déplacement $\boldsymbol{U} : I \to \mathbb{R}^{n_S}$ tel que

$$\mathbf{M}.\ddot{\boldsymbol{U}}(t) + \mathbf{K}.\boldsymbol{U}(t) = \boldsymbol{F}(t), \tag{1.15a}$$

$$\text{avec} \quad \boldsymbol{U}(0) = \boldsymbol{U}_0, \tag{1.15b}$$

$$\text{et} \quad \dot{\boldsymbol{U}}(0) = \boldsymbol{V}_0, \tag{1.15c}$$

où la matrice de masse est donnée par $\mathbf{M} = m(\boldsymbol{\phi}, \boldsymbol{\phi})$, la matrice de raideur par $\mathbf{K} = k(\boldsymbol{\phi}, \boldsymbol{\phi})$, et le vecteur des efforts extérieurs est donné par $\boldsymbol{F}(t) = \tilde{f}(\boldsymbol{\phi}; t)$. Les vecteurs $\boldsymbol{U}_0$ et $\boldsymbol{V}_0$ sont les coordonnées des champs initiaux $\tilde{u}_0$ et $\tilde{v}_0$ dans la base d'approximation. Les conventions $\dot{\boldsymbol{U}} = \frac{\mathrm{d}\boldsymbol{U}}{\mathrm{d}t}$ et $\ddot{\boldsymbol{U}} = \frac{\mathrm{d}^2\boldsymbol{U}}{\mathrm{d}t^2}$ sont utilisées.

Différentes méthodes peuvent être utilisées pour construire l'espace d'approximation. Le choix d'une méthode par rapport à une autre est dicté par un compromis entre la précision de l'approximation et le coût de calcul associé à l'assemblage des matrices **M** et **K**, et à la résolution d'un système linéaire de taille $n_S \times n_S$ à chaque instant (la meilleure méthode donnant l'approximation la plus précise à moindre coût). Dans ce manuscrit, on utilise la méthode des éléments finis, décrite par exemple dans [Hughes, 1987]. Le principe est de décomposer le domaine Ω en un ensemble de domaines élémentaires fermés de géométrie très simple. La base d'approximation est alors construite à l'aide de fonctions polynomiales dont le support est localisé sur un ou quelques éléments (voir l'Exemple 1.2). Le support des fonctions de base étant localisé, la plupart des coefficients des matrices **M** et **K** sont nuls, permettant ainsi de stocker ces matrices en mémoire même pour des valeurs de n_S très grandes, et d'utiliser des solveurs de système linéaire exploitant le caractère creux des matrices.

Exemple 1.2. (Construction de l'approximation éléments finis) Dans cet exemple, on décrit la construction de l'espace d'approximation $\mathscr{U}_h^S(\Omega;0)$ par la méthode des éléments finis. Dans une première étape, le domaine Ω est décomposé en N_S sous-domaines fermés de la façon suivante :

$$\Omega = \bigcup_{e=1}^{N_S} \Omega_e \quad \text{avec} \quad \Omega_e = [x_{e-1}, x_e]\,, \tag{1.16}$$

où $x_0 = 0$ et $x_{N_S} = L$. On considère ici un maillage uniforme, c'est-à-dire que l'on impose $x_e - x_{e-1} = h$ pour $e = 1, \cdots, N_S$. On construit ensuite, l'espace d'approximation $\mathscr{U}_h^S(\Omega)$ (sans se soucier des conditions aux limites), comme suit :

$$\mathscr{U}_h^S(\Omega) = \left\{ u \in \mathscr{U}^S(\Omega) \mid u \in \bigcup_{e=1}^{N_S} \mathscr{L}^p(\Omega_e) \right\}, \tag{1.17}$$

où $\mathscr{L}^p(\Omega_e)$ est l'espace des polynômes de Lagrange de degré p définis de Ω_e dans $\mathbb{R}$, dont la définition classique est la suivante [Prenter, 1975] :

$$\mathscr{L}^p(\Omega_e) = \left\{ u : \Omega_e \to \mathbb{R} \mid u(x) = \sum_{i=1}^{p+1} U_i\, l_i(x) \right\} \quad \text{avec} \quad l_i(x) = \prod_{\substack{j=1 \\ j \neq i}}^{p+1} \left(\frac{x - x_j}{x_i - x_j} \right), \tag{1.18}$$

où $\{x_1, \cdots, x_{p+1}\}$ forme une partition uniforme de Ω_e. Puis, on partitionne la base de $\mathscr{U}_h^S(\Omega)$ de la façon suivante :

$$u_h \in \mathscr{U}_h^S(\Omega) \Leftrightarrow u_h = \boldsymbol{\phi}.\boldsymbol{U} + \boldsymbol{\phi}^g.\boldsymbol{U}^g\,, \tag{1.19}$$

où le vecteur $\boldsymbol{\phi}$ contient les fonctions de la base d'approximation qui s'annulent sur les noeuds du maillage appartenant à $\partial\Omega_u$. On construit alors l'espace d'approximation $\mathscr{U}_h^S(\Omega;0)$ à partir de $\boldsymbol{\phi}$. Finallement, pour construire un champ cinématiquement admissible à g, on approche la fonction $\tilde{g}$ décrite dans la Remarque 1.4 avec les fonctions de formes contenues dans $\boldsymbol{\phi}^g$, soit $\tilde{g} \simeq \tilde{g}_h = \boldsymbol{\phi}^g.\boldsymbol{U}^g$ où les composantes du vecteur $\boldsymbol{U}^g$ sont identifiées en imposant $\tilde{g}_h(x_i) = g(x_i)$ pour tous les noeuds i appartenant à $\partial\Omega_u$.

Dans le cas d'un problème de propagation d'ondes, la précision de la base éléments finis dépend du nombre d'ondes k (voir l'Exemple 1.3). Pour les problèmes où le nombre d'ondes est grand (typiquement pour $k = 100$), une approximation par éléments finis linéaires requère une discrétisation très fine du domaine spatial, et n'est pas utilisable en pratique [Ihlenburg et Babuška, 1995]. Une solution consiste à augmenter le degré des polynômes utilisés pour construire la base éléments finis [Ihlenburg et Babuška, 1997]. De nombreuses méthodes ont été et sont développées afin d'améliorer la précision de l'espace d'approximation tout en ré-

duisant sa dimension. On cite ici à titre d'exemple : l'utilisation de fonctions $B-$spline [Hughes *et al.*, 2008], l'enrichissement de la base éléments finis à l'aide de fonctions harmoniques [Ham et Bathe, 2012] ou encore l'utilisation de méthodes de Galerkin discontinues en espace [Giorgiani *et al.*, 2013].

Remarque 1.5. *Dans le cas d'une sollicitation de choc, la réponse de la structure est caractérisée par de forts gradients par rapport à la variable spatiale. Comme on peut le voir sur la Figure 1.4, le support de cette perturbation est localisé, à un instant donné, sur une portion du domaine* Ω *dont la longueur est donnée par* $c\Delta T$ *(où on rappelle que* c *est la célérité des ondes dans le milieu et* ΔT *la durée caractéristique du choc). La discrétisation du domaine spatial doit alors être choisie de façon à pouvoir représenter cette perturbation. Il faut au minimum deux éléments linéaires*[3] *dans la longueur* $c\Delta T$ *pour « voir » la perturbation. Notons dès à présent que la même remarque peut être faite pour la discrétisation du domaine temporel. Pour une position donnée dans le domaine spatial, la durée de la perturbation est de* ΔT *et la durée du pas de temps ne doit donc pas excéder* $\Delta T/2$.

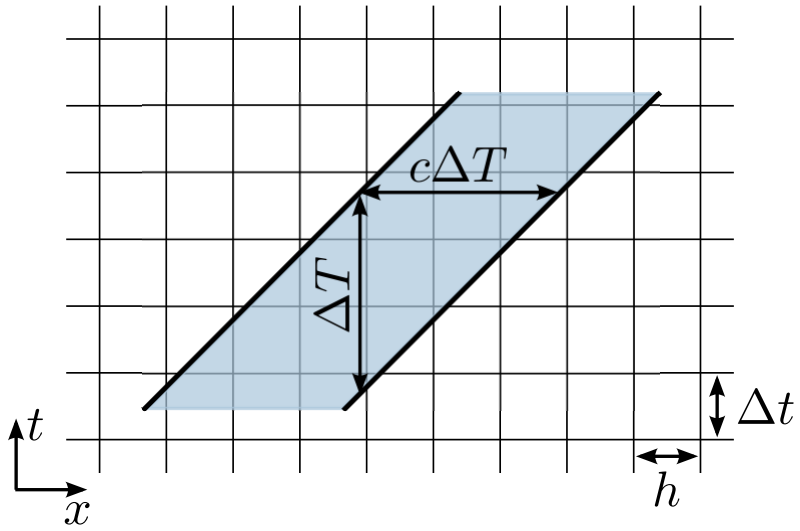

FIGURE 1.4: Propagation d'une onde sur un maillage espace-temps. La zone bleutée représente une zone de fort gradient par rapport aux variables spatiale et temporelle.

3. Pour des approximations de degré élevé, il faut au moins deux subdivisions d'un élément dans la longueur $c\Delta T$. Ceci permet d'utiliser un pas h de discrétisation plus grand que dans le cas d'élément linéaire.

Exemple 1.3. (Précision de l'approximation spatiale) La précision de l'approximation spatiale peut être jugée en évaluant la capacité de la base d'approximation à approcher la solution de l'équation d'Helmholtz, pour différentes valeurs du nombre d'ondes. Dans le cas unidimensionnel décrit dans l'Exemple 1.1, cette équation s'obtient en postulant un champ de déplacement de la forme $w(x)e^{i\omega t}$, où $w(x)$ et ω sont des inconnues. En remplaçant cette expression dans l'équation des ondes (1.7) et en divisant par $e^{i\omega t}$, on obtient alors l'équation d'Helmholtz :

$$\frac{\mathrm{d}^2 w}{\mathrm{d}x^2} + k^2 w = 0\,, \tag{1.20}$$

où $k = \frac{\omega}{c}$ est le nombre d'ondes. Dans le cas particulier de l'Exemple 1.1, $w(x)$ doit vérifier les conditions aux limites $w(0) = 0$ et $w(L) = 0$ (voir la Remarque 1.4). La solution générale de l'équation (1.20) est alors donnée par $w_n(x) = \sin(\frac{\omega_n}{c}x)$ avec $\omega_n = \frac{n\pi c}{L}$ pour $n = 1, 2, \cdots$. Le scalaire ω_n est appelé la pulsation propre de la structure et le champ $w_n(x)$ est le mode propre de vibration associé.

On peut alors chercher une approximation de $w(x)$ dans un espace d'approximation éléments finis de dimension n_S. On note $w_h(x) = \phi(x).\boldsymbol{W}$ cette approximation avec $\boldsymbol{W} \in \mathbb{R}^{n_S}$, et ω_h la pulsation propre associée. En multipliant l'équation (1.20) par une fonction test cinématiquement admissible à zéro, puis en intégrant sur le domaine Ω et en appliquant la formule de Green, on obtient le problème aux valeurs propres suivant :

$$(\mathbf{K} - \omega_h^2 \mathbf{M}).\boldsymbol{W} = 0\,. \tag{1.21}$$

où les matrices $\mathbf{K}$ et $\mathbf{M}$ sont les matrices de raideur et de masse. Les solutions $\boldsymbol{W}_n$ et ω_{hn}^2 de l'équation (1.21) sont les vecteurs et valeurs propres de l'opérateur $\mathbf{M}^{-1}.\mathbf{K}$. En résolvant numériquement ce problème aux valeurs propres, on peut comparer les valeurs exactes des pulsations ω_n avec leurs approximations ω_{hn} pour $n = 1, \cdots, n_S$.

Les résultats sont présentés sur la Figure 1.5. Pour les premières pulsations propres (pour $n < \frac{n_S}{2}$), on observe une convergence à l'ordre $2p$ de ω_{hn} vers ω_n (voir le cas représenté sur la figure de droite qui correspond à $n/n_S = 0.05$ sur la figure de gauche). La figure de gauche montre que l'erreur dépend de la pulsation propre considéré. Plus la pulsation propre est élevée et plus l'erreur est importante. De plus, pour les pulsations propres les plus élevées, on observe que l'approximation est d'autant moins précise que le degré p de l'approximation éléments finis est grand. Ceci traduit le fait que l'approximation éléments finis introduit des vecteurs propres associés aux hautes fréquences, qui n'ont pas de réalité physique.

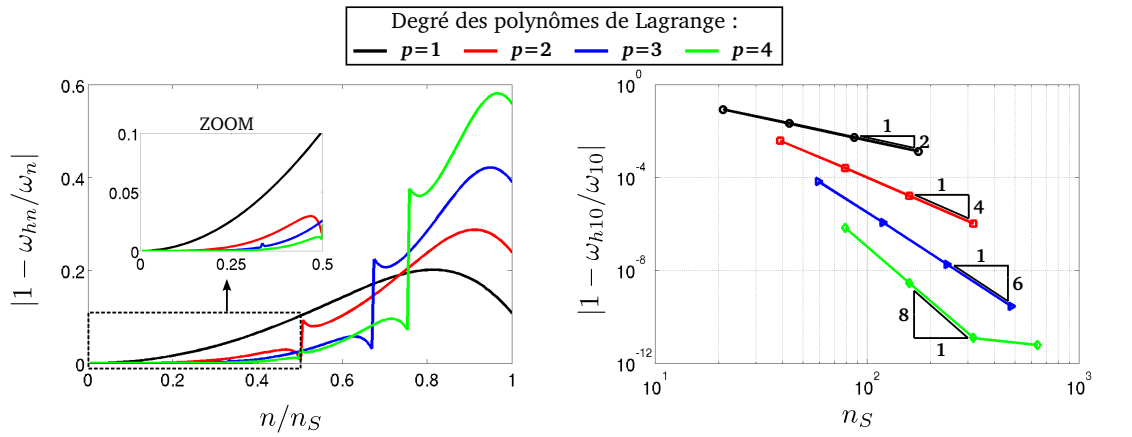

FIGURE 1.5: Erreur entre la pulsation propre ω_n et son approximation ω_{hn} pour différentes valeurs de n (à gauche) et convergence de l'approximation ω_{h10} vers ω_{10} lorsque $h = 1/n_S \to 0$ (à droite).

1.3 Approximation du problème en temps

Dans cette section, on montre comment résoudre le problème semi-discrétisé en espace à l'aide d'une méthode incrémentale en temps. On présente tout d'abord les schémas classiques d'intégration en temps, puis on décrit différentes méthodes éléments finis en temps.

1.3.1 Schémas d'intégration en temps

Dans le cas des schémas d'intégration en temps, on cherche une approximation du vecteur $\boldsymbol{U}(t)$ en un nombre fini d'instants $t \in \{t_i = i\Delta t \text{ pour } i = 0, \cdots, N_T\}$ de tel sorte que $t_0 = 0$ et $t_{N_T} = T$ et où Δt est l'incrément de temps (pris ici constant). L'idée est alors d'approcher le vecteur déplacement et ses dérivées à l'instant t_i à l'aide de leurs valeurs calculées à des instants antérieurs $t_{i-1}, t_{i-2}, \cdots$. Les méthodes diffèrent sur la façon de construire ces approximations et de résoudre l'équation de mouvement à l'instant t_i. On pourra consulter la deuxième partie du livre de [Hughes, 1987] pour une présentation détaillée des méthodes classiques. On trouvera dans la référence [Mahjoubi *et al.*, 2011], un formalisme général permettant d'implémenter de façon unifiée un ensemble de familles de schémas numériques, incluant les schémas les plus récents.

Schéma de Newmark

Dans ce manuscrit, on utilise le schéma d'intégration de Newmark (introduit dans [Newmark, 1959]) qui est souvent choisi comme référence. En pratique, ce schéma est l'un des plus utilisés dans les codes industriels. L'idée est d'utiliser un développement limité de Taylor des vecteurs déplacement et vitesse. L'approximation de Newmark est ainsi donnée par les développements suivants des vecteurs déplacement et vitesse, pris à l'instant $t_i = t_{i-1} + \Delta t$,

$$\boldsymbol{U}(t_i) = \boldsymbol{U}(t_{i-1}) + \Delta t \dot{\boldsymbol{U}}(t_{i-1}) + \tfrac{\Delta t^2}{2} \ddot{\boldsymbol{U}}(t_{i-1}) + \beta \Delta t^2 \big(\ddot{\boldsymbol{U}}(t_i) - \ddot{\boldsymbol{U}}(t_{i-1})\big), \tag{1.22a}$$

$$\dot{\boldsymbol{U}}(t_i) = \dot{\boldsymbol{U}}(t_{i-1}) + \Delta t \ddot{\boldsymbol{U}}(t_{i-1}) + \gamma \Delta t \big(\ddot{\boldsymbol{U}}(t_i) - \ddot{\boldsymbol{U}}(t_{i-1})\big), \tag{1.22b}$$

où les constantes β et γ sont des paramètres associés à l'approximation du reste des développements limités. En introduisant les approximations (1.22a) et (1.22b) dans l'équation de mouvement prise à l'instant t_i, soit

$$\mathbf{M}.\ddot{\boldsymbol{U}}(t_i) + \mathbf{K}.\boldsymbol{U}(t_i) = \boldsymbol{F}(t_i), \tag{1.23}$$

on obtient alors une formule de récurrence qui peut être initialisée à l'aide des conditions initiales et de l'équation de mouvement prise à l'instant initial :

$$\boldsymbol{U}(t_0) = \boldsymbol{U}_0, \tag{1.24a}$$

$$\dot{\boldsymbol{U}}(t_0) = \boldsymbol{V}_0, \tag{1.24b}$$

$$\text{et} \quad \ddot{\boldsymbol{U}}(t_0) = \mathbf{M}^{-1}.(\boldsymbol{F}(t_0) - \mathbf{K}.\boldsymbol{U}(t_0)). \tag{1.24c}$$

Cette formule de récurrence est généralement écrite en prévilégiant le vecteur accélération.

Schéma 1.1. Le schéma de Newmark consiste à calculer $\boldsymbol{U}(t_i), \dot{\boldsymbol{U}}(t_i)$ et $\ddot{\boldsymbol{U}}(t_i)$ en répétant les étapes suivantes pour $i = 1, \cdots, N_T$:

1. l'étape de prédiction,

$$\boldsymbol{U}^p(t_i) = \boldsymbol{U}(t_{i-1}) + \Delta t \dot{\boldsymbol{U}}(t_{i-1}) + \Delta t^2 (\tfrac{1}{2} - \beta) \ddot{\boldsymbol{U}}(t_{i-1}), \tag{1.25a}$$

$$\dot{\boldsymbol{U}}^p(t_i) = \dot{\boldsymbol{U}}(t_{i-1}) + \Delta t (1 - \gamma) \ddot{\boldsymbol{U}}(t_{i-1}), \tag{1.25b}$$

2. l'étape de résolution,

$$\big(\mathbf{M} + \beta \Delta t^2 \mathbf{K}\big).\ddot{\boldsymbol{U}}(t_i) = \boldsymbol{F}(t_i) - \mathbf{K}.\boldsymbol{U}^p(t_i), \tag{1.25c}$$

3. l'étape de correction,

$$\boldsymbol{U}(t_i) = \boldsymbol{U}^p(t_i) + \beta \Delta t^2 \ddot{\boldsymbol{U}}(t_i), \tag{1.25d}$$

$$\dot{\boldsymbol{U}}(t_i) = \dot{\boldsymbol{U}}^p(t_i) + \gamma \Delta t \ddot{\boldsymbol{U}}(t_i), \tag{1.25e}$$

où les vecteurs $\boldsymbol{U}(t_0), \dot{\boldsymbol{U}}(t_0)$ et $\ddot{\boldsymbol{U}}(t_0)$ sont donnés par les conditions initiales (1.24a) et (1.24b) et l'équation d'équilibre (1.24c).

Les schémas d'intégration en temps sont classiquement analysés en termes de stabilité, de dissipation des hautes fréquences, de précision, de procédure d'initialisation ou encore de coût de calcul [Hilber et Hughes, 1978]. Le comportement du schéma de Newmark vis-à-vis de ces propriétés est illustré dans l'Exemple 1.4. Pour les problèmes de choc, la dissipation des hautes fréquences est une propriété importante. En effet, la discrétisation spatiale de la structure par la méthode des éléments finis, introduit des modes propres de vibration associés aux hautes fréquences, qui n'ont pas de réalité physique (voir l'Exemple 1.3). Or pour les problèmes de choc, ces modes sont sollicités de façon non négligeable, et ils sont à l'origine d'oscillations parasites hautes fréquences dans la solution temporelle du problème semi-discrétisé. Il s'avère alors nécessaire d'introduire un amortissement numérique dans le schéma d'intégration en temps afin de supprimer la contribution des modes hautes fréquences, tout en conservant la contribution des modes de plus basses fréquences, et ceci sans dégrader l'ordre de convergence du schéma. C'est typiquement dans ce cas que le schéma de Newmark est mis en défaut. Aussi, de nombreuses méthodes (voir l'état de l'art dans [Hulbert, 2004]) ont été développées afin d'améliorer cet aspect. Parmi celles-ci, les méthodes éléments finis en temps ont de nombreux avantages (voir la comparaison de leurs propriétés avec celles du schéma de Newmark dans l'Exemple 1.4).

1.3.2 Méthodes éléments finis en temps

Les méthodes éléments finis en temps sont basées sur une formulation faible (en temps) du problème semi-discrétisé. Différentes méthodes variationnelles peuvent être utilisées pour construire cette formulation faible (on pourra consulter [Aharoni et Bar-Yoseph, 1992] et [Cannarozzi et Mancuso, 1995] pour différents exemples). Dans ce manuscrit, on s'intéresse plus particulièrement aux méthodes de Galerkin discontinues en temps à un champ (déplacement) et deux champs (déplacement-vitesse), introduites pour les problèmes hyperboliques du second ordre par [Hughes et Hulbert, 1988, Hulbert et Hughes, 1990, Hulbert, 1992]. L'implémentation de ces méthodes a notamment été détaillée par [Li et Wiberg, 1996, Li et Wiberg, 1998]. On notera également que les méthodes de Galerkin discontinues en temps peuvent être vues comme un cas particulier d'un schéma plus général construit à partir d'éléments finis étendus en temps [Rethore *et al.*, 2005].

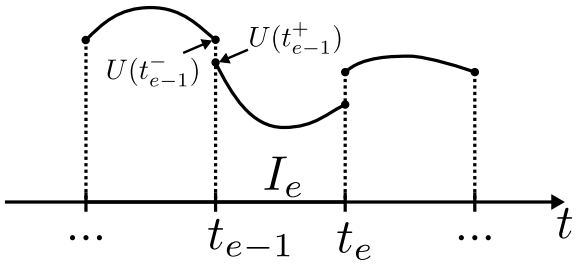

FIGURE 1.6: Exemple de fonction continue par morceaux.

Méthode de Galerkin discontinue en temps à un champ

Les méthodes de Galerkin discontinues en temps [4] sont basées sur une discrétisation du domaine temporel par intervalles de temps ouverts. Le domaine I est décomposé comme suit :

$$I = \bigcup_{e=1}^{N_T} \check{I}_e \quad \text{avec} \quad \check{I}_e =]\, t_{e-1}, t_e\,[, \tag{1.26}$$

où $t_0 = 0$ et $t_{N_T} = T$. On considère une partition uniforme du domaine temporel, c'est-à-dire $t_e - t_{e-1} = \Delta t$ pour $e = 1, \cdots, N_T$. Chaque composante $U_i(t)$ du vecteur déplacement $\boldsymbol{U}(t)$ est alors cherchée dans l'espace des fonctions continues par morceaux définies de I dans $\mathbb{R}$. Cet espace, noté [5] $\mathscr{U}^{\check{T}}(I)$, est défini par :

$$\mathscr{U}^{\check{T}}(I) = \left\{ u \in \bigcup_{e=1}^{N_T} \mathscr{U}^{T}(\check{I}_e) \right\}, \tag{1.27}$$

où $\mathscr{U}^{T}(I)$ est l'espace des fonctions définies de I dans $\mathbb{R}$, continues et suffisamment régulières. Le vecteur déplacement (ou sa dérivée temporelle) peut donc être discontinue à l'instant t_e. On utilise les notations $\boldsymbol{U}(t_e^{\pm}) = \lim_{\varepsilon \to 0^+} \boldsymbol{U}(t \pm \varepsilon)$. La continuité entre les intervalles de temps est alors imposée de façon faible.

Problème 1.4. La formulation faible associée à la méthode de Galerkin discontinue en temps à un champ consiste à trouver le vecteur déplacement $\boldsymbol{U} \in \left(\mathscr{U}^{\check{T}}\right)^{n_S}$ tel que $\forall \boldsymbol{U}^\star \in \left(\mathscr{U}^{\check{T}}\right)^{n_S}$, on ait :

$$B_e^{TDG-U}\left(\boldsymbol{U}^\star, \boldsymbol{U}\right) = L_e^{TDG-U}\left(\boldsymbol{U}^\star\right), \quad \text{pour } e = 1, \cdots, N_T, \tag{1.28a}$$

$$\begin{aligned} \text{avec} \quad B_e^{TDG-U}\left(\boldsymbol{U}^\star, \boldsymbol{U}\right) &= \int_{\check{I}_e} \dot{\boldsymbol{U}}^\star(t).\left(\mathbf{M}.\ddot{\boldsymbol{U}}(t) + \mathbf{K}.\boldsymbol{U}(t)\right) \mathrm{dt} \\ &\quad + \dot{\boldsymbol{U}}^\star(t_{e-1}^+).\mathbf{M}.\dot{\boldsymbol{U}}(t_{e-1}^+) + \boldsymbol{U}^\star(t_{e-1}^+).\mathbf{K}.\boldsymbol{U}(t_{e-1}^+), \end{aligned} \tag{1.28b}$$

$$\begin{aligned} \text{et} \quad L_e^{TDG-U}\left(\boldsymbol{U}^\star\right) &= \int_{\check{I}_e} \dot{\boldsymbol{U}}^\star(t).\boldsymbol{F}(t)\ \mathrm{dt} \\ &\quad + \dot{\boldsymbol{U}}^\star(t_{e-1}^+).\mathbf{M}.\dot{\boldsymbol{U}}(t_{e-1}^-) + \boldsymbol{U}^\star(t_{e-1}^+).\mathbf{K}.\boldsymbol{U}(t_{e-1}^-), \\ &\quad \text{pour } e = 2, \cdots, N_T, \end{aligned} \tag{1.28c}$$

$$\begin{aligned} \text{et} \quad L_1^{TDG-U}\left(\boldsymbol{U}^\star\right) &= \int_{\check{I}_1} \dot{\boldsymbol{U}}^\star(t).\boldsymbol{F}(t)\ \mathrm{dt} \\ &\quad + \dot{\boldsymbol{U}}^\star(t_0^+).\mathbf{M}.\boldsymbol{V}_0 + \boldsymbol{U}^\star(t_0^+).\mathbf{K}.\boldsymbol{U}_0\,. \end{aligned} \tag{1.28d}$$

4. On utilisera l'acronyme anglais « TDG » pour Time Discontinuous Galerkin.

5. L'exposant « T » dans la notation des espaces fonctionnels indique des fonctions définies sur le domaine temporel I. L'exposant « $\check{T}$ » indique des fonctions continues par morceaux. On notera $\mathscr{U}^T$ ou $\mathscr{U}^{\check{T}}$ (sans mentionner les détails) lorsqu'il n'y a pas d'ambiguïté sur la définition de l'espace considéré.

Dans cette formulation, les termes intégrales dans les équations (1.28b), (1.28c) et (1.28d) permettent d'imposer l'équation de mouvement sur chaque intervalle de temps $\check{I}_e$ pour $e = 1, \cdots, N_T$. La continuité du vecteur déplacement, ainsi que de sa dérivée temporelle, est imposée à chaque instant t_{e-1} avec les autres termes. En particulier, les conditions initiales en déplacement et vitesse sont imposées de façon faible.

Une approximation de la solution du Problème 1.4 est ensuite obtenue en introduisant une base d'approximation éléments finis en temps. Pour cela, l'espace $\mathscr{U}^{\check{T}}$ est remplacé par l'espace de dimension finie $\mathscr{U}^{\check{T}}_{\Delta t}$ défini par :

$$\mathscr{U}^{\check{T}}_{\Delta t}(I) = \left\{ u \in \mathscr{U}^{\check{T}}(I) \mid u \in \bigcup_{e=1}^{N_T} \mathscr{L}^p(\check{I}_e) \right\}, \tag{1.29}$$

où $\mathscr{L}^p(\check{I}_e)$ est l'espace des polynômes de Lagrange [6] de degré p définies de $\check{I}_e$ dans $\mathbb{R}$. La dimension de l'espace d'approximation est donnée par $n_T = \dim(\mathscr{U}^{\check{T}}_{\Delta t}) = (p+1)N_T$. Le vecteur déplacement $\boldsymbol{U}(t)$ est alors approché sous la forme $\boldsymbol{U}(t) \simeq \sum_{i=1}^{n_T} \boldsymbol{U}_i \check{\psi}_i(t)$ où $[\check{\psi}_1, \cdots, \check{\psi}_{n_T}]$ est la base éléments finis (discontinus) en temps. On utilise la même base d'approximation en temps pour $\boldsymbol{U}(t)$, $\boldsymbol{U}^\star(t)$ et $\boldsymbol{F}(t)$ (et donc la même numérotation). On associe à chaque élément $\check{I}_e$, le vecteur $\boldsymbol{l}_e = [l_1, \cdots, l_{p+1}]$ contenant les fonctions de forme locales [7]. La formulation faible étant écrite sur chaque intervalle de temps, on peut construire un schéma incrémental, en résolvant les équations (1.28a) l'une après l'autre, pour $e = 1, \cdots, N_T$. En supposant que l'équation $e-1$ a été résolu, on peut exprimer, à partir de l'équation e, les vecteurs (inconnus) $[\boldsymbol{U}_{(p+1)(e-1)+1}, \cdots, \boldsymbol{U}_{(p+1)e}]$ en fonction des vecteurs (connus) $[\boldsymbol{U}_{(p+1)(e-2)+1}, \cdots, \boldsymbol{U}_{(p+1)(e-1)}]$. On aboutit alors à un système linéaire dont la résolution permet de passer à l'incrément suivant.

Schéma 1.2. Le schéma incrémental associé à la méthode de Galerkin discontinue en temps à un champ, construit avec une base éléments finis de degré p (on note $n = p+1$), consiste à calculer $\boldsymbol{U}_{n*(e-1)+1}, \cdots, \boldsymbol{U}_{n*e}$ en répétant pour $e = 2, \cdots, N_T$

$$\begin{bmatrix} (1+a_{11})\mathbf{K}+b_{11}\mathbf{M} & \cdots & a_{1n}\mathbf{K}+b_{1n}\mathbf{M} \\ \vdots & \ddots & \vdots \\ a_{n1}\mathbf{K}+b_{n1}\mathbf{M} & \cdots & a_{nn}\mathbf{K}+b_{nn}\mathbf{M} \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{U}_{n*(e-1)+1} \\ \vdots \\ \boldsymbol{U}_{n*e} \end{bmatrix} = \begin{bmatrix} \tilde{\boldsymbol{F}}_1 \\ \vdots \\ \tilde{\boldsymbol{F}}_n \end{bmatrix} + \begin{bmatrix} c_{11}\mathbf{M} & \cdots & c_{1n}\mathbf{M}+\mathbf{K} \\ \vdots & \ddots & \vdots \\ c_{n1}\mathbf{M} & \cdots & c_{nn}\mathbf{M} \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{U}_{n*(e-2)+1} \\ \vdots \\ \boldsymbol{U}_{n*(e-1)} \end{bmatrix},$$

où $\mathbf{a} = \int_{t_{e-1}}^{t_e} \dot{\boldsymbol{l}}_e \otimes \boldsymbol{l}_e \, \mathrm{d}t$, $\mathbf{b} = \int_{t_{e-1}}^{t_e} \ddot{\boldsymbol{l}}_e \otimes \dot{\boldsymbol{l}}_e \, \mathrm{d}t + \dot{\boldsymbol{l}}_e(t_{e-1}) \otimes \dot{\boldsymbol{l}}_e(t_{e-1})$, $\mathbf{c} = \dot{\boldsymbol{l}}_e(t_{e-1}) \otimes \dot{\boldsymbol{l}}_{e-1}(t_{e-1})$ et $\tilde{\boldsymbol{F}}_i = \sum_{j=1}^{n} a_{ij} \boldsymbol{F}_{n*(e-1)+j}$ pour $i = 1, \cdots, n$. Le schéma incrémental est initialisé par

$$\begin{bmatrix} (1+a_{11})\mathbf{K}+b_{11}\mathbf{M} & \cdots & a_{1n}\mathbf{K}+b_{1n}\mathbf{M} \\ \vdots & \ddots & \vdots \\ a_{n1}\mathbf{K}+b_{n1}\mathbf{M} & \cdots & a_{nn}\mathbf{K}+b_{nn}\mathbf{M} \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{U}_1 \\ \vdots \\ \boldsymbol{U}_n \end{bmatrix} = \begin{bmatrix} \tilde{\boldsymbol{F}}_1 + d_1\mathbf{M}.\boldsymbol{V}_0 + \mathbf{K}.\boldsymbol{U}_0 \\ \vdots \\ \tilde{\boldsymbol{F}}_n + d_n\mathbf{M}.\boldsymbol{V}_0 \end{bmatrix}, \text{ où } \boldsymbol{d} = \dot{\boldsymbol{l}}_1(t_0).$$

6. Cet espace est défini de la même façon que dans le cas d'une approximation spatiale, voir l'équation (1.18).

7. Sur chaque élément $\check{I}_e$, on a $\check{\psi}_{(p+1)(e-1)+i} = l_i$ pour $i = 1, \cdots, p+1$, et $\check{\psi}_i = 0$ pour $i \notin [(p+1)(e-1)+1, (p+1)(e)]$ (voir la Figure 1.7).

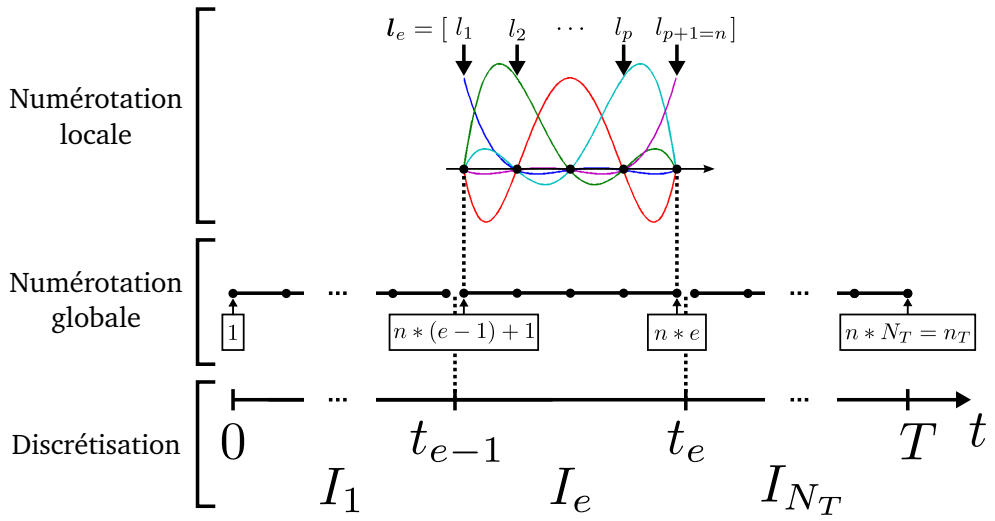

FIGURE 1.7: Implémentation de la méthode de Galerkin discontinue en temps.

Méthode de Galerkin discontinue en temps à deux champs

Dans la méthode de Galerkin discontinue en temps à deux champs, le vecteur vitesse est considéré comme un champ à part entière. Comme pour la méthode à un champ, chaque composante $V_i(t)$ du vecteur vitesse $\boldsymbol{V}(t)$ est cherchée dans l'espace des fonctions continues par morceaux, noté $\mathscr{U}^{\check{T}}$. L'égalité entre la dérivée temporelle du vecteur déplacement $\boldsymbol{U}(t)$ et le vecteur vitesse $\boldsymbol{V}(t)$ est alors imposée de façon faible sur chaque intervalle de temps.

Problème 1.5. La formulation faible associée à la méthode de Galerkin discontinue en temps à deux champs, consiste à trouver les vecteurs déplacement et vitesse $(\boldsymbol{U},\boldsymbol{V}) \in (\mathscr{U}^{\check{T}})^{n_S} \times (\mathscr{U}^{\check{T}})^{n_S}$ tel que $\forall (\boldsymbol{U}^\star,\boldsymbol{V}^\star) \in (\mathscr{U}^{\check{T}})^{n_S} \times (\mathscr{U}^{\check{T}})^{n_S}$, on ait :

$$B_e^{TDG-UV}\big((\boldsymbol{U}^\star,\boldsymbol{V}^\star),(\boldsymbol{U},\boldsymbol{V})\big) = L_e^{TDG-UV}\big((\boldsymbol{U}^\star,\boldsymbol{V}^\star)\big), \quad \text{pour } e = 1,\cdots,N_T, \tag{1.30a}$$

$$\begin{aligned} \text{avec} \quad B_e^{TDG-UV}\big((\boldsymbol{U}^\star,\boldsymbol{V}^\star),(\boldsymbol{U},\boldsymbol{V})\big) &= \int_{\check{I}_e} \boldsymbol{V}^\star(t).\big(\mathbf{M}.\dot{\boldsymbol{V}}(t) + \mathbf{K}.\boldsymbol{U}(t)\big)\,\mathrm{dt} \\ &+ \int_{\check{I}_e} \boldsymbol{U}^\star(t).\mathbf{K}.\big(\dot{\boldsymbol{U}}(t) - \boldsymbol{V}(t)\big)\,\mathrm{dt} \\ &+ \boldsymbol{V}^\star(t_{e-1}^+).\mathbf{M}.\boldsymbol{V}(t_{e-1}^+) + \boldsymbol{U}^\star(t_{e-1}^+).\mathbf{K}.\boldsymbol{U}(t_{e-1}^+), \end{aligned} \tag{1.30b}$$

$$\text{et} \quad L_e^{TDG-UV}\big((\boldsymbol{U}^\star,\boldsymbol{V}^\star)\big) = \int_{\check{I}_e} \boldsymbol{V}^\star(t).\boldsymbol{F}(t)\,\mathrm{dt} + \boldsymbol{V}^\star(t_{e-1}^+).\mathbf{M}.\boldsymbol{V}(t_{e-1}^-) + \boldsymbol{U}^\star(t_{e-1}^+).\mathbf{K}.\boldsymbol{U}(t_{e-1}^-), \quad \text{pour } e=2,\cdots,N_T, \tag{1.30c}$$

$$\text{et} \quad L_1^{TDG-UV}\big(\{\boldsymbol{U}^\star,\boldsymbol{V}^\star\}\big) = \int_{\check{I}_1} \boldsymbol{V}^\star(t).\boldsymbol{F}(t)\,\mathrm{dt} + \boldsymbol{V}^\star(t_0^+).\mathbf{M}.\boldsymbol{V}_0 + \boldsymbol{U}^\star(t_0^+).\mathbf{K}.\boldsymbol{U}_0\,. \tag{1.30d}$$

Le second terme intégral dans (1.30b) permet d'imposer l'égalité entre la dérivée temporelle du vecteur déplacement et le vecteur vitesse de façon faible, sur chaque intervalle $\check{I}_e$. Les autres termes sont similaires à ceux de la formulation à un champ. On introduit ensuite une base d'approximation éléments finis en temps pour approcher la solution du Problème 1.5. On peut alors choisir des espaces d'approximation différents pour le vecteur déplacement et le vecteur vitesse. Cependant, les propriétés de la méthode ne sont pas améliorées par un tel choix [Hulbert, 1992]. On choisit donc en pratique les mêmes espaces d'approximation pour $\boldsymbol{U}(t)$ et $\boldsymbol{V}(t)$ (c'est-à-dire l'espace $\mathscr{U}_{\Delta t}^{\check{T}}$ décrit précédemment), et on procède comme pour la méthode à un champ pour construire le schéma incrémental.

Schéma 1.3. Le schéma incrémental associé à la méthode de Galerkin discontinue en temps à deux champs, construit avec la même base éléments finis de degré p (on note $n=p+1$) pour les vecteurs déplacement et vitesse, consiste à calculer $\boldsymbol{U}_{n*(e-1)+1},\cdots,\boldsymbol{U}_{n*e}$ et $\boldsymbol{V}_{n*(e-1)+1},\cdots,\boldsymbol{V}_{n*e}$ en répétant pour $e=2,\cdots,N_T$

$$\begin{bmatrix} (1+a_{11})\mathbf{K} & \cdots & a_{1n}\mathbf{K} & -b_{11}\mathbf{K} & \cdots & -b_{1n}\mathbf{K} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ a_{n1}\mathbf{K} & \cdots & a_{nn}\mathbf{K} & -b_{n1}\mathbf{K} & \cdots & -b_{nn}\mathbf{K} \\ b_{11}\mathbf{K} & \cdots & b_{1n}\mathbf{K} & (1+a_{11})\mathbf{M} & \cdots & a_{1n}\mathbf{M} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ b_{n1}\mathbf{K} & \cdots & b_{nn}\mathbf{K} & a_{n1}\mathbf{M} & \cdots & a_{nn}\mathbf{M} \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{U}_{n*(e-1)+1} \\ \vdots \\ \boldsymbol{U}_{n*e} \\ \boldsymbol{V}_{n*(e-1)+1} \\ \vdots \\ \boldsymbol{V}_{n*e} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \vdots \\ \mathbf{0} \\ \tilde{\boldsymbol{F}}_1 \\ \vdots \\ \tilde{\boldsymbol{F}}_n \end{bmatrix} + \begin{bmatrix} \mathbf{0} & \cdots & \mathbf{K} & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{M} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{U}_{n*(e-2)+1} \\ \vdots \\ \boldsymbol{U}_{n*(e-1)} \\ \boldsymbol{V}_{n*(e-2)+1} \\ \vdots \\ \boldsymbol{V}_{n*(e-1)} \end{bmatrix},$$

où $\mathbf{a}=\int_{t_{e-1}}^{t_e} \boldsymbol{l}_e \otimes \dot{\boldsymbol{l}}_e\,\mathrm{dt}$, $\mathbf{b}=\int_{t_{e-1}}^{t_e} \boldsymbol{l}_e \otimes \boldsymbol{l}_e\,\mathrm{dt}$ et $\tilde{\boldsymbol{F}}_i=\sum_{j=1}^{n} b_{ij}\boldsymbol{F}_{n*(e-1)+j}$ pour $i=1,\cdots,n$.
Le schéma incrémental est initialisé par

$$\begin{bmatrix} (1+a_{11})\mathbf{K} & \cdots & a_{1n}\mathbf{K} & -b_{11}\mathbf{K} & \cdots & -b_{1n}\mathbf{K} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ a_{n1}\mathbf{K} & \cdots & a_{nn}\mathbf{K} & -b_{n1}\mathbf{K} & \cdots & -b_{nn}\mathbf{K} \\ b_{11}\mathbf{K} & \cdots & b_{1n}\mathbf{K} & (1+a_{11})\mathbf{M} & \cdots & a_{1n}\mathbf{M} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ b_{n1}\mathbf{K} & \cdots & b_{nn}\mathbf{K} & a_{n1}\mathbf{M} & \cdots & a_{nn}\mathbf{M} \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{U}_1 \\ \vdots \\ \boldsymbol{U}_n \\ \boldsymbol{V}_1 \\ \vdots \\ \boldsymbol{V}_n \end{bmatrix} = \begin{bmatrix} \mathbf{K}.\boldsymbol{U}_0 \\ \vdots \\ \mathbf{0} \\ \mathbf{M}.\boldsymbol{V}_0+\tilde{\boldsymbol{F}}_1 \\ \vdots \\ \tilde{\boldsymbol{F}}_n \end{bmatrix}.$$

Méthode de Galerkin continue en temps à deux champs

Une autre stratégie consiste à imposer la continuité entre les intervalles de temps de façon forte. On obtient alors une méthode de Galerkin continue en temps, par exemple décrite par [French et Peterson, 1996], ou plus récemment dans une version modifiée par [Idesman, 2007]. On considère ici la méthode classique. Le domaine temporel est décomposé en N_T intervalles fermés, comme suit :

$$I = \bigcup_{e=1}^{N_T} I_e \quad \text{avec} \quad I_e = [t_{e-1}, t_e], \tag{1.31}$$

où $t_0 = 0$ et $t_{N_T} = T$. On choisit $t_e - t_{e-1} = \Delta t$ pour $e = 1, \cdots, N_T$. La principale différence avec les méthodes de Galerkin discontinues en temps est que les conditions initiales sont imposées de façon forte : chaque composante du vecteur déplacement est cherchée dans l'espace des fonctions (continues et régulières) qui vérifient a priori la condition initiale en déplacement (et de façon similaire pour le vecteur vitesse). Cet espace, noté $\mathscr{U}^T(I; g_0)$, est appelé l'espace des fonctions cinématiquement admissibles (en temps) à la condition initiale g_0. Il est défini comme suit :

$$\mathscr{U}^T(I; g_0) = \left\{ u \in \mathscr{U}^T(I) \mid u(0) = g_0 \right\}. \tag{1.32}$$

Ainsi, le vecteur déplacement $\boldsymbol{U}(t)$ est cherché dans l'espace $\mathscr{U}^T(I; \boldsymbol{U}_0) = \left\{ \boldsymbol{U}(t) \mid U_i \in \mathscr{U}^T(I; U_{i0}) \text{ pour } i = 1, \cdots, n_S \right\}$ où le scalaire U_{i0} est la i-ème composante du vecteur déplacement initial $\boldsymbol{U}_0$. Le vecteur vitesse $\boldsymbol{V}(t)$ est cherché de façon similaire dans l'espace $\mathscr{U}^T(I; \boldsymbol{V}_0)$ où $\boldsymbol{V}_0$ est le vecteur vitesse initiale. On choisit alors les fonctions tests $\boldsymbol{U}^\star(t)$ et $\boldsymbol{V}^\star(t)$ dans l'espace $\mathscr{U}^T(I; \boldsymbol{0})$ des fonctions cinématiquement admissibles à zéros. Dans ce cadre, la formulation faible du problème consiste simplement à imposer, sur le domaine temporel I, l'équation de mouvement, et l'égalité entre le vecteur vitesse et la dérivée temporelle du vecteur déplacement.

Problème 1.6. La formulation faible associée à la méthode de Galerkin continue en temps à deux champs consiste à trouver le couple $(\boldsymbol{U}, \boldsymbol{V}) \in \mathscr{U}^T(I; \boldsymbol{U}_0) \times \mathscr{U}^T(I; \boldsymbol{V}_0)$ tel que $\forall (\boldsymbol{U}^\star, \boldsymbol{V}^\star) \in \mathscr{U}^T(I; \boldsymbol{0}) \times \mathscr{U}^T(I; \boldsymbol{0})$, on ait :

$$B^{TG-UV}\big((\boldsymbol{U}^\star, \boldsymbol{V}^\star), (\boldsymbol{U}, \boldsymbol{V})\big) = L^{TG-UV}\big((\boldsymbol{U}^\star, \boldsymbol{V}^\star)\big), \tag{1.33a}$$

$$\begin{aligned} \text{avec} \quad B^{TG-UV}\big((\boldsymbol{U}^\star, \boldsymbol{V}^\star), (\boldsymbol{U}, \boldsymbol{V})\big) &= \int_I \boldsymbol{V}^\star(t).\big(\mathbf{M}.\dot{\boldsymbol{V}}(t) + \mathbf{K}.\boldsymbol{U}(t)\big)\,\mathrm{dt} \\ &\quad + \int_I \boldsymbol{U}^\star(t).\mathbf{K}.\big(\dot{\boldsymbol{U}}(t) - \boldsymbol{V}(t)\big)\,\mathrm{dt}, \end{aligned} \tag{1.33b}$$

$$\text{et} \quad L^{TG-UV}\big((\boldsymbol{U}^\star, \boldsymbol{V}^\star)\big) = \int_I \boldsymbol{V}^\star(t).\boldsymbol{F}(t)\,\mathrm{dt}. \tag{1.33c}$$

Une approximation de la solution du Problème 1.6 peut alors être obtenue en introduisant un espace d'approximation par éléments finis (continus) en temps. La démarche est la même que dans le cas d'une approximation spatiale [8]. Cependant, cette démarche aboutit à la résolution d'un système linéaire de dimension $n \times n$ avec $n = n_S n_T$ où n_S et n_T sont les dimensions des espaces d'approximation spatiale et temporele respectivement. En pratique, ce système est bien trop grand pour qu'il soit possible de le résoudre avec un solveur classique [9]. Aussi, afin de permettre une résolution incrémentale, le Problème 1.6 est reformulé sur chaque intervalle de temps I_e, en supposant que les conditions initiales sont les valeurs des vecteurs déplacement et vitesse calculées à la fin de l'intervalle de temps I_{e-1}, à l'incrément précédent. En notant $\boldsymbol{U}(t_{e-1}) = \boldsymbol{U}_{e-1}$ et $\boldsymbol{V}(t_{e-1}) = \boldsymbol{V}_{e-1}$ les nouvelles conditions initiales sur l'intervalle I_e, le problème consiste alors à trouver le couple $(\boldsymbol{U}, \boldsymbol{V})\big|_{I_e} \in \mathscr{U}^T(I_e; \boldsymbol{U}_{e-1}) \times \mathscr{U}^T(I_e; \boldsymbol{V}_{e-1})$ tel que $\forall (\boldsymbol{U}^\star, \boldsymbol{V}^\star) \in \mathscr{U}^T(I_e; \boldsymbol{0}) \times \mathscr{U}^T(I_e; \boldsymbol{0})$, on ait pour $e = 1, \cdots, N_T$:

$$B_e^{TG-UV}\big((\boldsymbol{U}^\star, \boldsymbol{V}^\star), (\boldsymbol{U}, \boldsymbol{V})\big) = L_e^{TG-UV}\big((\boldsymbol{U}^\star, \boldsymbol{V}^\star)\big), \tag{1.34a}$$

$$\begin{aligned} \text{avec} \quad B_e^{TG-UV}\big((\boldsymbol{U}^\star, \boldsymbol{V}^\star), (\boldsymbol{U}, \boldsymbol{V})\big) &= \int_{I_e} \boldsymbol{V}^\star(t).\big(\mathbf{M}.\dot{\boldsymbol{V}}(t) + \mathbf{K}.\boldsymbol{U}(t)\big)\,\mathrm{dt} \\ &\quad + \int_{I_e} \boldsymbol{U}^\star(t).\mathbf{K}.\big(\dot{\boldsymbol{U}}(t) - \boldsymbol{V}(t)\big)\,\mathrm{dt}, \end{aligned} \tag{1.34b}$$

$$\text{et} \quad L_e^{TG-UV}\big((\boldsymbol{U}^\star, \boldsymbol{V}^\star)\big) = \int_{I_e} \boldsymbol{V}^\star(t).\boldsymbol{F}(t)\,\mathrm{dt}. \tag{1.34c}$$

La restriction du vecteur déplacement sur l'intervalle de temps I_e, pour $e = 1, \cdots, N_T$, est ensuite approchée sous la forme $\boldsymbol{U}|_{I_e} \simeq \sum_{i=1}^{p+1} \boldsymbol{U}_{p(e-1)+i-1} l_i(t)$ où $[l_1, \cdots, l_{p+1}]$ est la base de $\mathscr{L}^p(I_e)$ qui sert à définir l'espace d'approximation éléments finis en temps. La numérotation est illustrée sur la Figure 1.8. Les fonctions de forme l_i sont partitionnées [10] comme suit : $l_e^g = l_1$ et $\boldsymbol{l}_e = [l_2, \cdots, l_{p+1}]$. La même base d'approximation est utilisée pour le vecteur vitesse $\boldsymbol{V}(t)$ et le vecteur des efforts extérieurs $\boldsymbol{F}(t)$. Les fonctions tests $\boldsymbol{U}^\star(t)$ et $\boldsymbol{V}^\star(t)$ sont approchées sur la base $\boldsymbol{l}_e$.

8. En suivant la Remarque 1.4, une fonction de $\mathscr{U}^T(I; g_0)$ est construite comme la somme d'une fonction (inconnue) de $\mathscr{U}^T(I; 0)$ et d'une fonction $\tilde{g}_0$ définie sur I et égale à g_0 à l'instant initial. Puis un espace d'approximation $\mathscr{U}^T_{\Delta t}(I; 0)$ est construit en partitionnant la base de l'espace $\mathscr{U}^T_{\Delta t}(I)$ de la même façon que dans l'Exemple 1.2.

9. Un des objectifs de cette thèse est justement de permettre, grâce à la méthode de séparation de variables, de trouver une approximation de la solution de ce système linéaire « espace-temps », à moindre coût.

10. Sur chaque intervalle I_e, les fonctions $[l_1, \cdots, l_{p+1}]$ permettent de construire l'espace d'approximation $\mathscr{U}^S_{\Delta t}(I_e)$. Les fonctions $[l_2, \cdots, l_{p+1}]$ permettent de construire l'espace d'approximation $\mathscr{U}^S_{\Delta t}(I_e; 0)$ et la fonction l_1 permet de construire la fonction $\tilde{g}$ définie sur I_e et qui est égale à la condition initiale à l'instant t_{e-1}.

Schéma 1.4. Le schéma incrémental associé à la méthode de Galerkin continue en temps à deux champs, construit avec une base éléments finis de degré p pour les vecteurs déplacement et vitesse, consiste à calculer $\boldsymbol{U}_{p*(e-1)+1},\cdots,\boldsymbol{U}_{p*e}$ et $\boldsymbol{V}_{p*(e-1)+1},\cdots,\boldsymbol{V}_{p*e}$ en répétant pour $e=1,\cdots,N_T$

$$\begin{bmatrix} a_{11}\mathbf{K} & \cdots & a_{1p}\mathbf{K} & -b_{11}\mathbf{K} & \cdots & -b_{1p}\mathbf{K} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ a_{p1}\mathbf{K} & \cdots & a_{pp}\mathbf{K} & -b_{p1}\mathbf{K} & \cdots & -b_{pp}\mathbf{K} \\ b_{11}\mathbf{K} & \cdots & b_{1p}\mathbf{K} & a_{11}\mathbf{M} & \cdots & a_{1p}\mathbf{M} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ b_{p1}\mathbf{K} & \cdots & b_{pp}\mathbf{K} & a_{p1}\mathbf{M} & \cdots & a_{pp}\mathbf{M} \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{U}_{p*(e-1)+1} \\ \vdots \\ \boldsymbol{U}_{p*e} \\ \boldsymbol{V}_{p*(e-1)+1} \\ \vdots \\ \boldsymbol{V}_{p*e} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \vdots \\ \mathbf{0} \\ \tilde{\boldsymbol{F}}_1 \\ \vdots \\ \tilde{\boldsymbol{F}}_p \end{bmatrix} - \begin{bmatrix} c_1\mathbf{K} & -d_1\mathbf{K} \\ \vdots & \vdots \\ c_p\mathbf{K} & -d_p\mathbf{K} \\ d_1\mathbf{K} & c_1\mathbf{M} \\ \vdots & \vdots \\ d_p\mathbf{K} & c_p\mathbf{M} \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{U}_{p*(e-1)} \\ \boldsymbol{V}_{p*(e-1)} \end{bmatrix}, \tag{1.35}$$

où $\mathbf{a}=\int_{t_{e-1}}^{t_e} \boldsymbol{l}_e \otimes \dot{\boldsymbol{l}}_e \,\mathrm{dt}$, $\mathbf{b}=\int_{t_{e-1}}^{t_e} \boldsymbol{l}_e \otimes \boldsymbol{l}_e \,\mathrm{dt}$, $\mathbf{c}=\int_{t_{e-1}}^{t_e} \boldsymbol{l}_e l_e^g \,\mathrm{dt}$, $\mathbf{d}=\int_{t_{e-1}}^{t_e} \boldsymbol{l}_e l_e^g \,\mathrm{dt}$ et $\tilde{\boldsymbol{F}}_i = d_i \boldsymbol{F}_{p*(e-1)} + \sum_{j=1}^{p} b_{ij} \boldsymbol{F}_{p*(e-1)+j}$ pour $i=1,\cdots,p$.

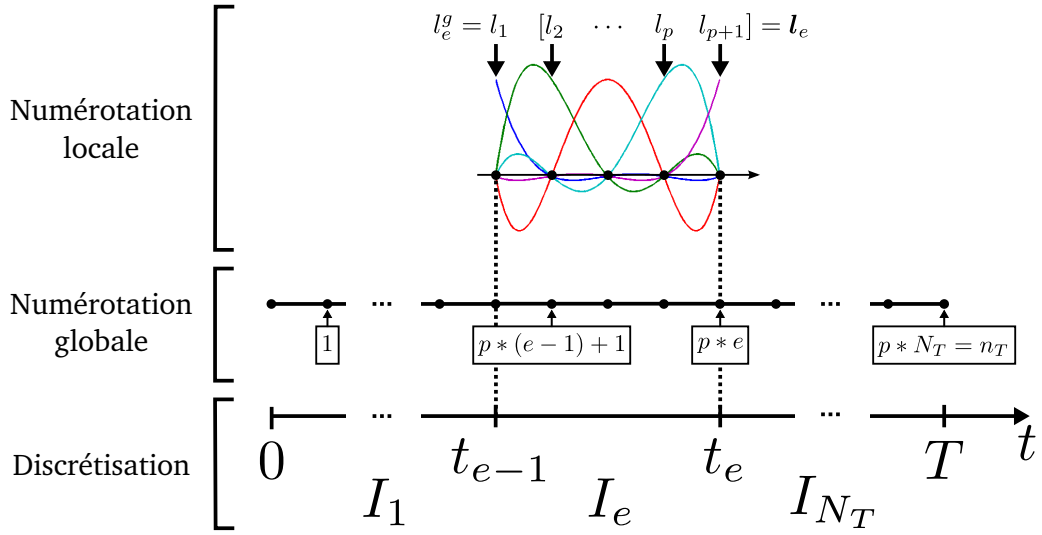

FIGURE 1.8: Implémentation de la méthode de Galerkin continue en temps.

Exemple 1.4. (Propriétés des méthodes d'approximation en temps) Dans cet exemple, on compare les propriétés des méthodes d'approximation en temps introduites précédemment, à savoir : le schéma de Newmark, les méthodes de Galerkin discontinues en temps à un champ (TDG-U) et deux champs (TDG-UV), et la méthode de Galerkin continue en temps à deux champs (TG-UV). Les méthodes de Galerkin en temps sont discrétisées à l'aide d'éléments finis dont les fonctions de forme sont des

polynômes de Lagrange de degré p. Pour les formulations en déplacement-vitesse, le même degré d'approximation est utilisé pour chaque champ. On note « TDG Pp », « TDG Pp-Pp » et « TG Pp-Pp » les méthodes TDG-U, TDG-UV et TG-UV, respectivement, avec approximation polynomiale de degré p. On compare les propriétés de stabilité, de dissipation des hautes fréquences, d'ordre de convergence et de coût de calcul associées à chacune de ces méthodes.

Équation modale. Pour étudier les propriétés des méthodes d'approximation en temps, il est commode de remplacer l'équation de mouvement (1.15a) par un système d'équations découplées. Ce système d'équations est obtenu en remplaçant l'espace $\mathscr{U}_h^S$ par l'espace construit à partir des modes propres de la structure. Les propriétés d'orthogonalité des modes par rapport aux matrices de masse et de raideur, permettent alors d'écrire le problème semi-discrétisé sous la forme d'un système de n_S équations découplées, où chaque équation décrit le mouvement d'une coordonnée du vecteur déplacement dans la base modale. En notant $q(t)$ la coordonnée modale du vecteur déplacement, associé à la pulsation propre ω_h, on aboutit à une équation de la forme :

$$\ddot{q}(t) + \omega_h^2 q(t) = f(t) \quad \text{avec} \quad \begin{cases} q(0) = q_0 \\ \dot{q}(0) = \dot{q}_0 \end{cases}, \tag{1.36}$$

où f, q_0 et $\dot{q}_0$ sont respectivement les coordonnées (associées à la pulsation propre ω_h) des vecteurs de forces extérieures, de déplacement initial et vitesse initiale dans la base modale. L'équation (1.36) est appelée équation modale.

Stabilité. La stabilité d'un schéma numérique permet de garantir que la solution reste bornée à chaque instant t_i, notamment si des perturbations dues aux erreurs d'arrondis sont introduites durant la résolution. Un schéma est dit stable si le rayon spectral de sa matrice d'amplification est inférieur ou égal à un. La matrice d'amplification (notée **A**) est obtenue en exprimant le schéma incrémental sous la forme :

$$\boldsymbol{y}_i = \mathbf{A}.\boldsymbol{y}_{i-1} + \boldsymbol{b}_i\,, \tag{1.37}$$

où $\boldsymbol{y}_i$ est le vecteur d'état à l'incrément i. Ce vecteur d'état dépend de la méthode d'approximation en temps utilisée. Pour le schéma de Newmark, on prend $\boldsymbol{y}_i = [q(t_i), \dot{q}(t_i)]$, pour le schéma TDG-U $\boldsymbol{y}_i = [q(t_{i-1}^+), \cdots, q(t_i^-)]$, pour le schéma TDG-UV $\boldsymbol{y}_i = [q(t_i^-), \Delta t\dot{q}(t_i^-)]$ et pour le schéma TG-UV on prend $\boldsymbol{y}_i = [q(t_i), \Delta t\dot{q}(t_i)]$. Le rayon spectral de la matrice d'amplification est alors défini par $\rho(A) = \max_i(|\lambda_i(\mathbf{A})|)$ où les scalaires $\lambda_i(\mathbf{A})$ sont les valeurs propres de **A**. Finalement, pour étudier la stabilité du schéma, on représente le rayon spectral de la matrice d'amplification en fonction des valeurs de $\Delta t\,\omega_h/(2\pi)$, pour les différentes méthodes (voir la Figure 1.9).

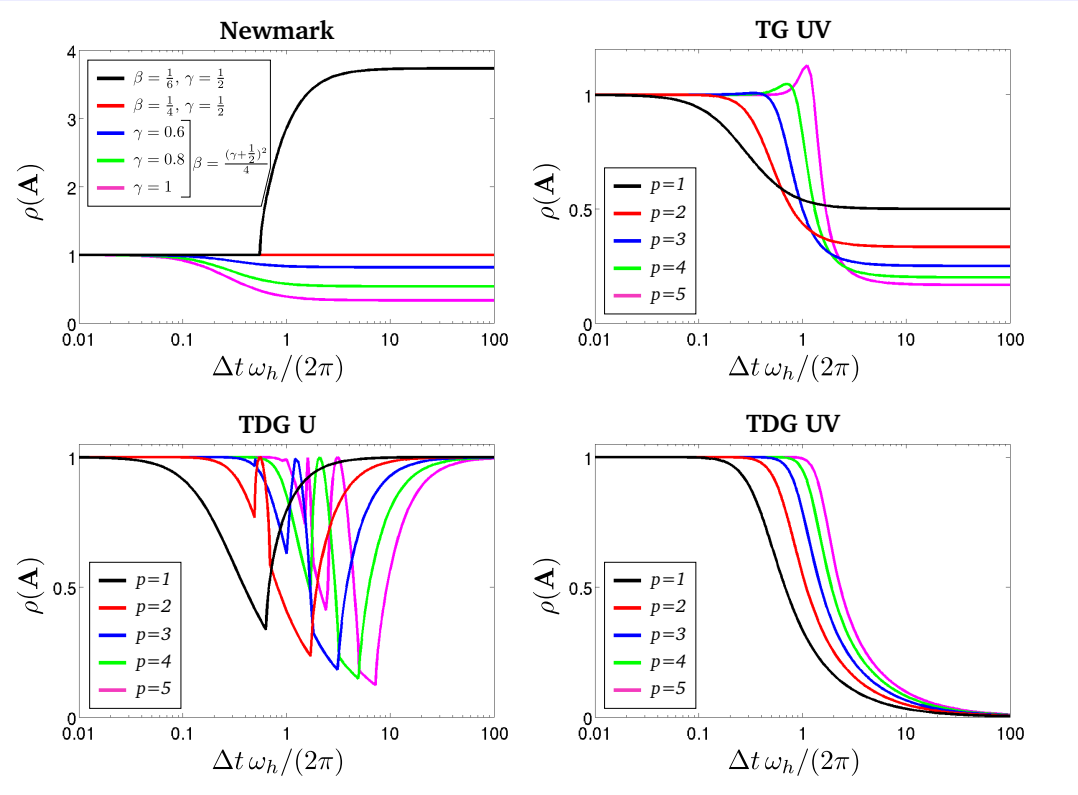

FIGURE 1.9: Rayon spectral de la matrice d'amplification en fonction du pas de temps et de la pulsation propre du problème semi-discrétisé.

Pour les méthodes TDG-U et TDG-UV avec $p = 1, \cdots, 5$, pour la méthode TG-UV avec $p = 1, 2$ et pour le schéma de Newmark avec $2\beta \geq \gamma \geq \frac{1}{2}$, on observe que le rayon spectral est inférieur ou égal à un, quelles que soient les valeurs de $\Delta t\,\omega_h/(2\pi)$. La stabilité du schéma est dite inconditionnelle. Pour la méthode TG-UV avec $p > 2$ et le schéma de Newmark avec $\gamma \geq \frac{1}{2}$ et $\beta < \frac{\gamma}{2}$, on observe que le rayon spectral est supérieur à un pour certaines valeurs de $\Delta t\,\omega_h/(2\pi)$. La stabilité du schéma est dite conditionnelle. Dans ce cas, pour que la solution converge, le pas de temps doit être choisi de façon à ce que $\Delta t\,\omega_h \leq \Omega_{\text{crit}}$ où Ω_{crit} est un seuil qui dépend des paramètres du schéma (et éventuellement de l'amortissement physique de la structure).

La plus grande valeur de ω_h peut être bornée par la pulsation propre du plus petit des éléments du maillage de la structure. Aussi, dans le cas où le maillage spatial contient de très petits éléments, l'utilisation d'un schéma conditionnellement stable peut nécessiter des pas de temps inutilement très faibles. Cependant, dans le cas d'un problème de choc, le choix du pas de temps est également contraint par la durée ΔT de la sollicitation (on doit pouvoir représenter correctement le chargement au cours du temps, voir la Remarque 1.5). Dans ce cas, l'utilisation d'un schéma conditionnellement stable n'est pas forcément désavantageux (le maillage spatial et le pas de temps

doivent être choisis de façon à pouvoir représenter des variations locales du champ de déplacement, qui nécessitent de toute façon un pas de temps très fin).

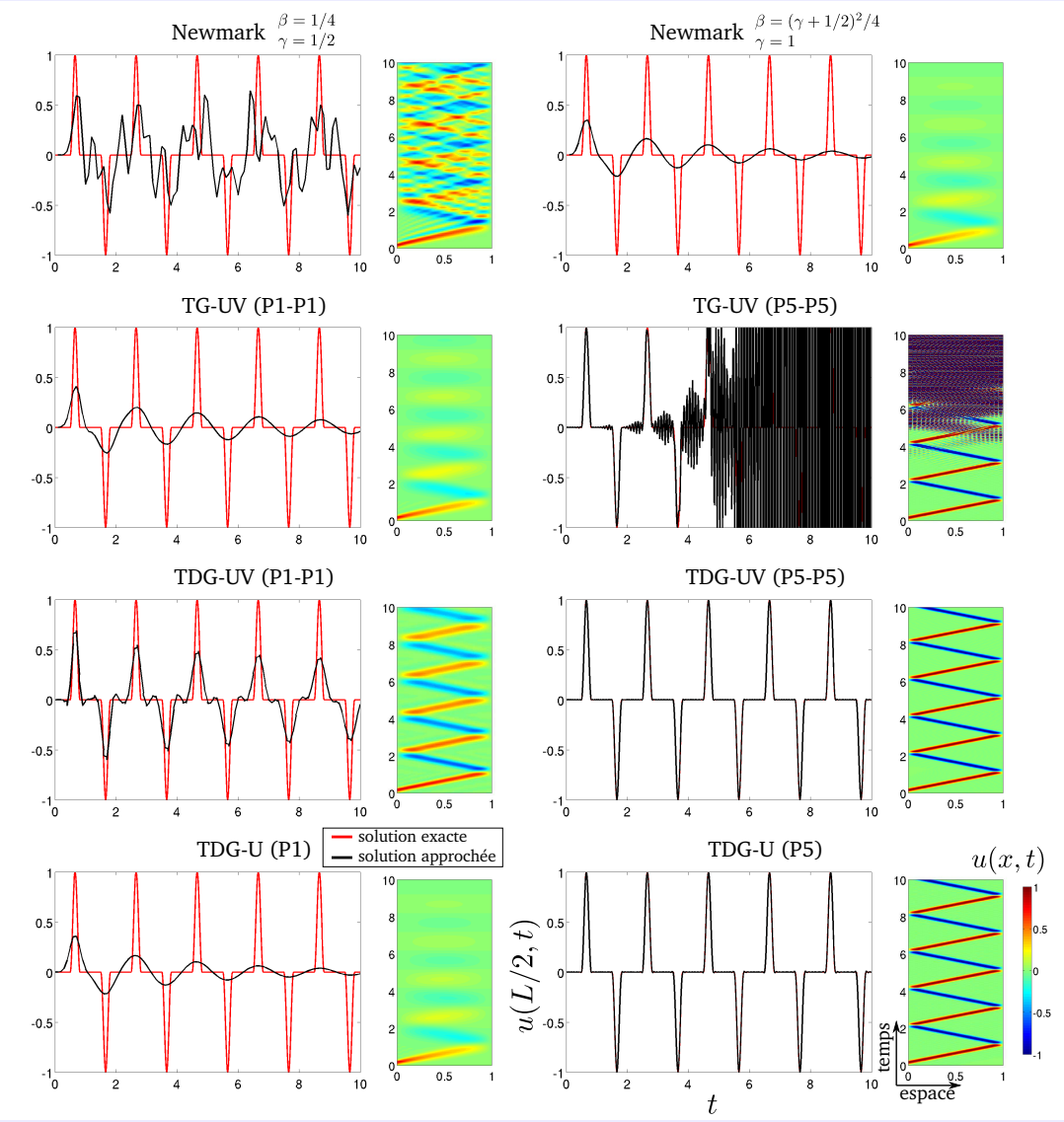

FIGURE 1.10: Champ de déplacement obtenu avec le même maillage espace-temps pour toutes les méthodes représentées (espace : $h = L/100$ (éléments de degré 5), temps : $\Delta t = T/100$). Le cas test est décrit dans l'Exemple 1.1 avec $\kappa = 10$.

Dissipation des hautes fréquences. Dans le cas des problèmes de choc, les modes propres associés aux vibrations hautes fréquences sont sollicités de façon non négligeable. Or, comme on l'a vu dans l'Exemple 1.3, l'utilisation de la méthode des éléments

finis (en espace) introduit des modes hautes fréquences qui ne sont pas physiques. Ces modes sont à l'origine d'oscillations parasites hautes fréquences dans la réponse temporelle du problème semi-discrétisé (ce phénomène est illustré sur la Figure 1.10). Ainsi, la méthode d'approximation en temps doit permettre de filtrer ces oscillations parasites, sans pour autant atténuer les modes de plus basses fréquences. Idéalement, cette propriété se traduit par un rayon spectral qui tend vers zéro lorsque $\Delta t\,\omega_h/(2\pi)$ tend vers l'infini et qui est égal à un lorsque $\Delta t\,\omega_h/(2\pi) \leq 1$.

Pour le schéma de Newmark, le choix des paramètres $\gamma > \frac{1}{2}$ et $\beta = (\gamma + \frac{1}{2})^2/4$ permet d'amortir les modes hautes fréquences (voir le rayon spectral pour $\Delta t\,\omega_h/(2\pi) \to \infty$ sur la Figure 1.9). Cependant, l'amortissement introduit pour les hautes fréquences a tendance à attenuer également les modes basses fréquences (voir le rayon spectral pour $\Delta t\,\omega_h/(2\pi) \in [0.1, 1]$). Comme on peut l'observer sur la Figure 1.9, la méthode TDG-UV présente un comportement quasiment idéal en terme de dissipation des hautes fréquences dès lors que l'on augmente le degré de l'approximation. La méthode TG-UV présente un comportement similaire à celui de la méthode TDG-UV. Cependant, pour un degré d'approximation supérieure à deux, le schéma devient instable. Aussi, seulement les schémas TG-P1P1 et TG-P2P2 peuvent être utilisés en pratique. Dans [Idesman, 2007], un amortissement numérique controlé est introduit pour pallier à ce problème. Enfin, la méthode TDG-U ne permet d'amortir que certaine bande de fréquence. L'ajout d'opérateurs de stabilisation dans la formulation faible permet d'obtenir un comportement proche de celui de la méthode TDG-UV [Hulbert, 1992].

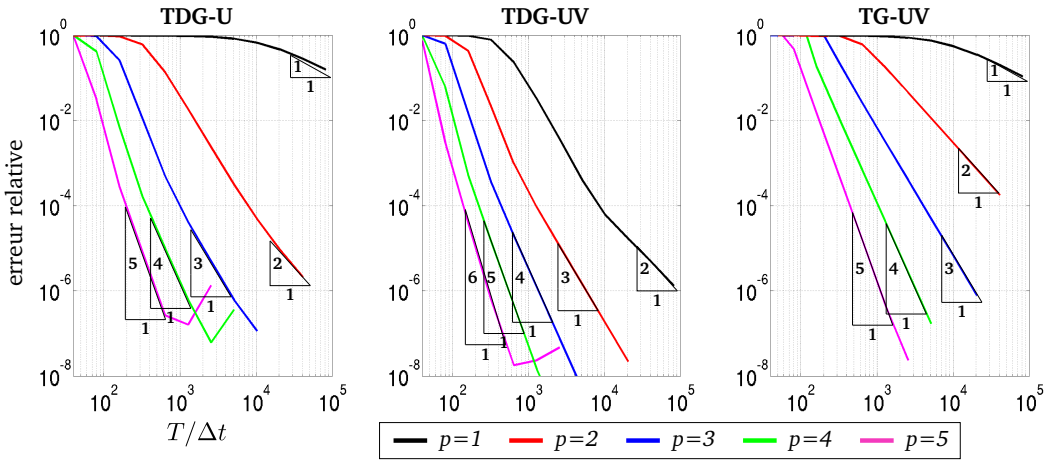

FIGURE 1.11: Erreur relative $\| q - q_{\Delta t} \|_{ener} / \| q \|_{ener}$ en fonction du pas de temps Δt, pour le problème modal avec condition initiale en déplacement ($t \in [0, T]$ avec $T = 10$ et $\omega = 10\pi$).

Ordre de convergence. Lorsque l'on introduit de l'amortissement numérique pour supprimer les hautes fréquences parasites, la difficulté est de conserver un ordre de convergence élevé de la méthode. Typiquement, le schéma de Newmark converge à l'ordre deux si et seulement si $\gamma = \frac{1}{2}$ et il converge à l'ordre un sinon. L'introduction d'amortissement numérique (pour les valeurs de $\gamma > \frac{1}{2}$) dégrade donc la convergence du schéma de Newmark. Pour les méthodes éléments finis en temps, une évaluation numérique de l'ordre de convergence est présentée sur la Figure 1.11. L'erreur est calculée avec la norme énergétique $\| \, \|_{ener}$ définie (avec l'équation modale) par $\| q \|^2_{ener} = \int_I \frac{1}{2}(k q^2 + m \dot{q}^2)$ dt où m et k sont la masse modale et la raideur modale, associées à la pulsation propre ω. On résout l'équation modale avec $f(t) = 0$, $q_0 = 1$ et $\dot{q}_0 = 0$ pour les différentes méthodes d'approximation en temps (la solution approchée est notée $q_{\Delta t}$). La solution exacte est donnée par $q(t) = \sin(\omega t)$. Puis, on trace $\| q - q_{\Delta t} \|_{ener} / \| q \|_{ener}$ pour différentes valeurs du pas de temps Δt. Avec la norme énergétique, on observe une convergence asymptotique à l'ordre p pour les méthodes TDG-U et TG-UV et à l'ordre $p+1$ pour la méthode TDG-UV.

Coût de calcul. Afin de comparer la complexité des différentes méthodes, on note $\mathbf{lin}(n)$ la complexité associée à la résolution d'un système linéaire de taille $n \times n$, et $\mathbf{mv}(n)$ la complexité associée au calcul d'un produit matrice-vecteur dont la matrice est de taille $n \times n$. On suppose que la matrice (du système linéaire que l'on doit résoudre à chaque incrément) à une largeur de bande (et un conditionnement) similaire pour les différentes méthodes.

Pour le schéma de Newmark, le cas le plus favorable est obtenu lorsque $\beta = 0$ et que la matrice de masse est diagonalisée. On parle dans ce cas de schéma explicite. Un schéma explicite nécessite de l'ordre de $N_T \, \mathbf{mv}(n_S)$ opérations pour obtenir la solution espace-temps. Dans le cas où $\beta \neq 0$, on parle de schéma implicite et un système linéaire doit être résolu à chaque instant. La compléxité du schéma implicite est de l'ordre de $N_T \, \mathbf{lin}(n_S)$, ce qui donne clairement l'avantage aux méthodes explicites. Cependant les méthodes implicites permettent d'utiliser un pas de temps plus large.

Les méthodes éléments finis en temps sont également des méthodes implicites. La taille du système linéaire à résoudre pour chaque intervalle de temps, augmente avec le degré de l'approximation. Ceci est le principal inconvénient de ces méthodes. Ainsi pour la méthode TDG-U, la complexité du schéma est de l'ordre de $N_T \, \mathbf{lin}\big((p+1)n_S\big)$, pour la méthode TDG-UV, elle est de l'ordre de $N_T \, \mathbf{lin}\big(2(p+1)n_S\big)$ et pour la méthode TG-UV de $N_T \, \mathbf{lin}\big(2pn_S\big)$. Néanmoins le nombre d'intervalles de temps N_T, nécessaire pour obtenir la solution à une précision donnée, est beaucoup plus faible pour les méthodes éléments finis de degré élevé que pour le schéma de Newmark (par exemple, pour le cas test représenté sur la Figure 1.10, il faut 23660 intervalles de temps avec le schéma de Newmark $\beta = 1/4$ et $\gamma = 1/2$ pour obtenir la solution avec une erreur rela-

tive inférieure à 10^{-2} (dans la norme énergétique), alors que seulement 273 intervalles de temps suffisent pour le schéma TDG P5-P5).

Finalement, on notera que des méthodes basées sur une résolution itérative du système linéaire à chaque intervalle de temps, ont été proposées pour réduire le coût de calcul associé aux méthodes éléments finis en temps de degré élévé (voir par exemple les stratégies proposées dans [Kunthong et Thompson, 2005] ou [Idesman, 2007] pour les méthodes TDG-UV et TG-UV respectivement). La complexité est alors ramenée à un ordre de grandeur de $N_T \xi \mathbf{lin}(n_S)$ où ξ est un nombre d'itérations (dans [Kunthong et Thompson, 2005], les valeurs $\xi = 2$ ou $\xi = 3$ sont suffisantes pour des approximations de degré un ou deux, cependant des valeurs de ξ plus grandes doivent être utilisées pour préserver les propriétés des méthodes TDG lorsque le degré de l'approximation augmente). La stratégie proposée dans la suite de ce manuscrit peut être vue comme une alternative à ces travaux.

1.4 Conclusion

Dans ce chapitre, la modélisation en variables espace-temps du problème d'élastodynamique a été présentée. Cette modélisation est basée sur une semi-discrétisation du problème en espace et une résolution incrémentale en temps. Les principales propriétés des méthodes d'approximation en espace et en temps ont été décrites. Notamment, la discrétisation spatiale par la méthode des éléments finis, introduit des modes hautes fréquences qui n'ont pas de réalité physique. Pour les problèmes de choc, ces modes sont excitées de façon non négligeable et introduisent des oscillations parasites dans la solution du problème semi-discrétisé. Aussi, les méthodes de Galerkin discontinues en temps, permettent de supprimer ces oscillations parasites tout en préservant un ordre de convergence élevé de l'approximation en temps.

Le potentiel des méthodes de Galerkin discontinues en temps de degré élévé est cependant limité par le coût de calcul associé au système linéaire qu'il est nécessaire de résoudre à chaque intervalle de temps. **Cette limite est clairement liée à la nature incrémentale de la stratégie de résolution.** Dans la suite de ce manuscrit, on propose une stratégie de résolution non-incrémentale qui doit permettre notamment, de réduire les coûts de calcul associés aux méthodes éléments finis en temps de degré élevé. La complexité de cette stratégie est de l'ordre de grandeur de $M\xi\left(\mathbf{lin}(n_S) + \mathbf{lin}(n_T)\right)$ où n_S est la dimension de l'espace d'approximation spatiale et n_T celle de l'espace d'approximation temporelle, ξ est un nombre d'itérations et M un nombre de modes espace-temps.

Toute la stratégie est basée sur une approximation du champ de déplacement (ou de vitesse) sous la forme d'une représentation à variables séparées espace-temps. Cette représentation peut être vue comme une somme de M modes espace-temps. La

question est donc de savoir quel est la valeur de M qui permette d'obtenir une bonne approximation de la solution d'un problème de dynamique transitoire. Répondre à cette question est l'objet du chapitre suivant.

Chapitre 2

Compression de données par séparation de variables espace-temps

Dans ce chapitre, on évalue l'efficacité de la méthode de séparation de variables espace-temps, en terme de compression de la mémoire nécessaire au stockage de la solution sur le domaine espace-temps, de problèmes académiques de dynamique transitoire.

Sommaire

2.1 Motivations

La modélisation classique des phénomènes de propagation d'ondes dans des structures complexes requère des discrétisations spatiales et temporelles très fines pour représenter le phénomène avec une précision raisonnable. Aussi, **le stockage des résultats devient rapidement un problème.** Supposons par exemple qu'un maillage espace-temps avec un million de degrés de liberté en espace et un million de pas de temps soit nécessaire pour simuler la réponse transitoire d'une structure. Alors, le stockage de la solution sur ce maillage espace-temps demanderait environ 7500 Go. Un tel espace mémoire ne paraît pas si impressionnant au vu des capacités actuelles des moyens informatiques et on peut penser qu'une telle mémoire sera abordable aisément dans une dizaine d'années. Cependant, la demande en moyen de calculs dépasse toujours les capacités disponibles à une époque donnée. Le réel enjeu est donc de permettre une meilleure exploitation des ressources disponibles.

Lorsque l'espace mémoire nécessaire pour stocker les résultats sur tout le domaine espace-temps est trop important, une solution consiste à ne sauvegarder qu'une partie des résultats (la partie complémentaire étant effacée au cours du calcul). Cette solution, bien que non satisfaisante puisqu'une partie des résultats est perdue, est adoptée dans tous les codes de calculs industriels. Ainsi, après avoir réalisé la mise en données, l'analyste doit sélectionner, avant de lancer la simulation, quels résultats post-traiter, sur quels degrés de liberté et à quels instants. Ce choix repose sur son expertise et sa connaissance du phénomène modélisé ou encore sur des résultats expérimentaux. Cependant, lorsque le phénomène modélisé est complexe ou mal connu, il s'avère difficile de savoir quelle partie de la structure sera la plus sollicitée et à quel instant, avant de lancer la simulation. Dans ce cas, pouvoir visualiser les résultats sur tout le domaine espace-temps est un réel avantage, ne serait-ce que pour déboguer un modèle. Compresser les résultats de la simulation est alors indispensable.

La séparation de variables s'avère une piste très prometteuse pour atteindre un tel objectif. Pour une grande variété de problèmes, l'approximation de la solution sous la forme d'une représentation à variables séparées permet de réduire la mémoire nécessaire au stockage des résultats de plusieurs ordres de grandeur [Chinesta *et al.*, 2011]. Cependant la précision d'une telle approximation dépend du problème considéré, et pour le problème de dynamique transitoire, l'intérêt d'une représentation à variables séparées espace-temps reste à démontrer. Dans ce chapitre, on se concentre sur cet aspect. On suppose que la solution (discrète) du problème est connue sur le domaine espace-temps. On définit alors la meilleure approximation (au sens d'un problème de minimisation) de cette solution sous la forme d'une représentation à variables séparées espace-temps. On évalue ensuite le gain mémoire, associé au stockage des résultats sous format séparé ou non, pour des problèmes académiques de dynamique transitoire.

Remarque 2.1. *La séparation de variables, vue comme outil de compression de données, nécessite de repenser les outils de visualisation des résultats. Le format utilisé pour stocker une représentation à variables séparées diffère des formats usuels et il est donc nécessaire d'adapter les outils existants à ce type de représentation [Bordeu, 2013]. Dans un logiciel standard, le post-traitement (par exemple le calcul du champ de contraintes à partir du champ de déplacement) et le stockage des résultats interviennent durant l'étape de résolution du problème, typiquement à la fin de chaque pas de temps. L'utilisation d'un outil efficace de compression de données permettrait de différencier complétement les étapes de résolution et de post-traitement. En autorisant le stockage du champ solution sur tout le domaine de calcul, on peut concevoir un logiciel de post-traitement « intelligent », qui ne se limite pas à la visualisation de données mais qui puisse calculer d'autres quantités, en temps réel, telles que des champs de déformation ou de contrainte à partir de la donnée d'un champ de déplacement sous format séparé. De telles modifications sont lourdes du point de vue développement logiciel. L'objectif de ce chapitre est d'évaluer l'intérêt de tels développements, sur des cas tests académiques de dynamique transitoire.*

2.2 Séparation de variables espace-temps

Soit $u(x,t)$ un champ scalaire, supposé connu, défini sur le domaine espace-temps $\Omega \times I$ et appartenant à un espace produit tensoriel $\mathscr{U}^S \otimes \mathscr{U}^T$. La méthode de séparation de variables espace-temps consiste à chercher une approximation, notée $u_M \in \mathscr{U}^S \otimes \mathscr{U}^T$, du champ u, sous la forme suivante [Ladevèze, 1999] :

$$\boxed{u(x,t) \simeq u_M(x,t) = \sum_{m=1}^{M} w_m(x)\,\lambda_m(t)}. \tag{2.1}$$

Chaque fonction $w_m \in \mathscr{U}^S$ est appelée un mode en espace et chaque fonction $\lambda_m \in \mathscr{U}^T$ est appelée un mode en temps. Le produit $w_m \lambda_m$ est appelé un mode espace-temps. Le nombre de modes espace-temps, noté M, est appelé le rang de l'approximation.

Exemple 2.1. (Intérêt en dimension finie) En pratique, le champ que l'on cherche à approximer sous la forme d'une représentation à variables séparées appartient à un espace de dimension finie. On note ce champ $u^{h,\Delta t}$ et $\mathscr{U}_h^S \otimes \mathscr{U}_{\Delta t}^T$ l'espace de dimension finie auquel il appartient. Ce champ peut s'exprimer dans une base de $\mathscr{U}_h^S \otimes \mathscr{U}_{\Delta t}^T$ construite par tensorisation de bases, c'est-à-dire

$$u^{h,\Delta t}(x,t) = \phi(x) \otimes \psi(t) \overset{D}{\cdot} \mathbf{U} \quad \text{avec} \quad \mathbf{U} \in \mathbb{R}^{n_S} \otimes \mathbb{R}^{n_T}, \tag{2.2}$$

où $\phi = [\phi_1, \cdots, \phi_{n_S}]$ est une base de $\mathscr{U}_h^S$ et $\psi = [\psi_1, \cdots, \psi_{n_T}]$ est une base de $\mathscr{U}_{\Delta t}^T$. Le produit « $\overset{D}{\cdot}$ » est le produit scalaire canonique entre tenseurs d'ordre D (ici $D = 2$) introduit dans l'Annexe A. Dans le cas où la base $\phi \otimes \psi$ est construite par la méthode

des éléments finis, le tenseur $\mathbf{U}$ contient la valeur du champ $u^{h,\Delta t}$ en tout noeud du maillage espace-temps, c'est-à-dire

$$\mathbf{U} = \begin{bmatrix} u^{h,\Delta t}(x_1,t_1) & \cdots & u^{h,\Delta t}(x_1,t_{n_T}) \\ \vdots & \ddots & \vdots \\ u^{h,\Delta t}(x_{n_S},t_1) & \cdots & u^{h,\Delta t}(x_{n_S},t_{n_T}) \end{bmatrix}. \tag{2.3}$$

C'est la mémoire nécessaire au stockage de $\mathbf{U}$ que l'on cherche à compresser à l'aide d'une approximation à variables séparées espace-temps. Cette approximation (noté $u_M^{h,\Delta t}$) appartient également à l'espace $\mathscr{U}_h^S \otimes \mathscr{U}_{\Delta t}^T$ et peut donc être exprimée sur la base $\phi \otimes \psi$ comme suit

$$u_M^{h,\Delta t}(x,t) = \phi(x) \otimes \psi(t) \underbrace{\overset{\mathrm{D}}{\cdot}\Big(\sum_{m=1}^{M} \boldsymbol{W}_m \otimes \boldsymbol{\Lambda}_m \Big)}_{\mathbf{U}_M} \quad \text{avec} \quad \begin{cases} \boldsymbol{W}_m \in \mathbb{R}^{n_S} \\ \boldsymbol{\Lambda}_m \in \mathbb{R}^{n_T} \end{cases}. \tag{2.4}$$

En regroupant (2.2) et (2.4) dans (2.1), on peut alors écrire la méthode de séparation de variables espace-temps sous forme discrète :

$$\boxed{\mathbf{U} \simeq \mathbf{U}_M = \sum_{m=1}^{M} \boldsymbol{W}_m \otimes \boldsymbol{\Lambda}_m}. \tag{2.5}$$

On comprend ainsi l'intérêt d'une approximation de $\mathbf{U}$ par une représentation à variables séparées : dans sa représentation complète, il faut $n_S n_T$ nombres réels pour stocker $\mathbf{U}$, alors que le stokage de sa représentation sous format séparé $\mathbf{U}_M$ demande $M(n_S + n_T)$ nombres réels. Dans le cas où $M << \frac{n_S n_T}{n_S + n_T}$, on aura besoin de beaucoup moins de mémoire pour stocker $\mathbf{U}_M$ que pour stocker $\mathbf{U}$. L'enjeu est donc de savoir s'il est possible d'approcher $\mathbf{U}$ de façon précise par $\mathbf{U}_M$ avec un rang M suffisamment faible devant $\frac{n_S n_T}{n_S + n_T}$.

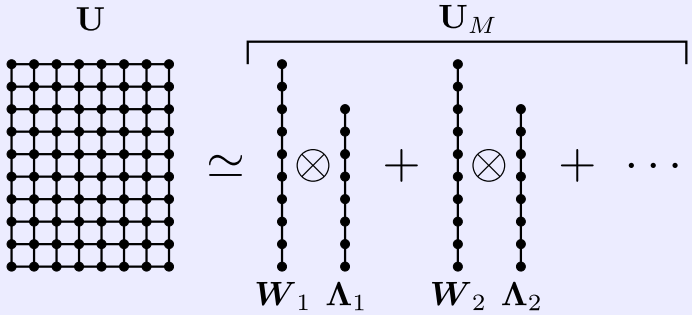

FIGURE 2.1: Approximation d'un tenseur $\mathbf{U}$ par un autre tenseur $\mathbf{U}_M$ de rang M stocké sous format séparé.

Afin d'évaluer s'il est possible de représenter de façon précise la solution d'un problème de dynamique transitoire avec une représentation à variables séparées espace-temps de faible rang, il est tout d'abord nécessaire de définir quelle est la meilleure approximation de rang M et comment la calculer.

2.2.1 Meilleure approximation de rang M

On cherche une approximation du champ u dans le sous-ensemble $\mathscr{R}_M$ des représentations à variables séparées espace-temps de rang M, défini par

$$\mathscr{R}_M = \left\{ u \in \mathscr{U}^S \otimes \mathscr{U}^T \mid u(x,t) = \sum_{m=1}^{M} w_m(x)\,\lambda_m(t) \text{ avec } w_m \in \mathscr{U}^S,\, \lambda_m \in \mathscr{U}^T \right\}. \quad (2.6)$$

Afin de définir la notion de meilleure approximation, il est nécessaire d'introduire une mesure de l'erreur entre le champ u et une décomposition de rang M appartenant à $\mathscr{R}_M$. Pour cela, on note $\langle , \rangle$ un produit scalaire sur $\mathscr{U}^S \otimes \mathscr{U}^T$ et $\| \ \|$ la norme associée. On définit alors la meilleure approximation de rang M comme solution du problème de minimisation suivant [Hackbusch, 2012] :

Définition 2.1. Soit $u \in \mathscr{U}^S \otimes \mathscr{U}^T$ un champ connu. La meilleure approximation u_M de u dans $\mathscr{R}_M$, par rapport à la norme $\| \ \|$, est définie par

$$u_M \in \arg\min_{u^\star \in \mathscr{R}_M} \| u - u^\star \| . \quad (2.7)$$

Remarque 2.2. *Dans le cas d'un espace de dimension finie, on note $\mathscr{R}_M^{h,\Delta t}$ le sous-ensemble des représentations à variables séparées espace-temps de rang M, défini par :*

$$\mathscr{R}_M^{h,\Delta t} = \left\{ u \in \mathscr{U}_h^S \otimes \mathscr{U}_{\Delta t}^T \mid u(x,t) = \phi(x) \otimes \psi(t) \overset{D}{\cdot} \mathbf{U} \text{ avec } \mathbf{U} \in \mathrm{R}_M \right\}, \quad (2.8)$$

où R_M est le sous-ensemble des décompositions espace-temps de rang M défini par :

$$\mathrm{R}_M = \left\{ \mathbf{U} \in \mathbb{R}^{n_S} \otimes \mathbb{R}^{n_T} \mid \mathbf{U} = \sum_{m=1}^{M} \boldsymbol{W}_m \otimes \boldsymbol{\Lambda}_m \text{ avec } \boldsymbol{W}_m \in \mathbb{R}^{n_S}, \boldsymbol{\Lambda}_m \in \mathbb{R}^{n_T} \right\}. \quad (2.9)$$

Remarque 2.3. *Lorsque la norme utilisée pour définir le problème de meilleure approximation vérifie la propriété de séparabilité suivante :*

$$\| w\,\lambda \| = \| w \|_S \| \lambda \|_T \quad , \forall w \in \mathscr{U}^S, \forall \lambda \in \mathscr{U}^T, \quad (2.10)$$

où $\| \ \|_S$ et $\| \ \|_T$ sont des normes définies sur $\mathscr{U}^S$ et $\mathscr{U}^T$ respectivement, alors la solution u_M du problème de meilleure approximation (2.7) est appelée la décomposition orthogonale propre (POD) de u (également appelée décomposition de Karhunen-Loève

(KLD) ou analyse en composantes principales (PCA) ou décomposition en valeur singulière (SVD), voir [Liang et al.*, 2002] pour une comparaison)* [1].

Remarque 2.4. *Si la propriété (2.10) n'est pas vérifiée, alors u_M est appelé la décomposition généralisée propre (PGD) de u (également appelée décomposition spectrale généralisée (GSD), voir [Nouy, 2007, Nouy, 2010a]).*

Remarque 2.5. *L'approximation u_M est appelée la décomposition « a posteriori » de u lorsque le champ u est connu. C'est le cas traité dans ce chapitre. Elle est appelée décomposition « a priori » lorsque le champ u n'est pas connu.*

2.2.2 Construction a posteriori

On détaille ici la construction a posteriori de la solution du problème de meilleure approximation (2.7) dans le cas où la norme $\|\ \|$ vérifie la propriété de séparabilité (2.10) et que le champ que l'on cherche à décomposer appartient à un espace de dimension finie. Dans ce cas, la meilleure approximation de rang M peut être construite avec la décomposition en valeurs singulières du tenseur $\mathbf{U} \in \mathbb{R}^{n_S} \otimes \mathbb{R}^{n_T}$. La forme discrète du problème (2.7) consiste à trouver $\mathbf{U}_M \in \mathrm{R}_M$ tel que

$$\mathbf{U}_M \in \arg \min_{\mathbf{U}^\star \in \mathrm{R}_M} \|\mathbf{U} - \mathbf{U}^\star\|, \tag{2.11}$$

où la représentation discrète de la norme $\|\ \|$ est définie sur $\mathbb{R}^{n_S} \otimes \mathbb{R}^{n_T}$ et associée au produit scalaire [2] $\langle\ ,\ \rangle$ tel que

$$\langle \mathbf{U}, \mathbf{V} \rangle = \mathbf{U} \cdot^{\mathrm{D}} (\mathbf{N}_S \otimes \mathbf{N}_T) \cdot^{\mathrm{D}} \mathbf{V}, \tag{2.12}$$

avec $\mathbf{N}_S = \langle \phi, \phi \rangle_S$ et $\mathbf{N}_T = \langle \psi, \psi \rangle_T$. Les matrices $\mathbf{N}_S \in \mathbb{R}^{n_S} \otimes \mathbb{R}^{n_S}$ et $\mathbf{N}_T \in \mathbb{R}^{n_T} \otimes \mathbb{R}^{n_T}$ sont symétriques définies positives.

Cas de la norme canonique

Lorsque les matrices $\mathbf{N}_S$ et $\mathbf{N}_T$ sont des matrices identités, la représentation discrète de la norme $\|\ \|$ correspond à la norme canonique [3] pour les tenseurs d'ordre deux, que l'on notera $\|\ \|_2$. Dans ce cas, le théorème d'approximation de Schmidt [4] [Schmidt, 1907] montre qu'une solution $\mathbf{U}_M$ du problème de meilleure approximation (2.11) est donnée par la SVD de $\mathbf{U}$ tronquée au rang M.

1. Les acronymes sont donnés en anglais
2. Pour tout $u, v \in \mathscr{U}_h^S \otimes \mathscr{U}_{\Delta t}^T$, on a $\langle u, v \rangle = \langle \phi \otimes \psi \cdot^{\mathrm{D}} \mathbf{U}, \phi \otimes \psi \cdot^{\mathrm{D}} \mathbf{V} \rangle = \mathbf{U} \cdot^{\mathrm{D}} \langle \phi, \phi \rangle_S \otimes \langle \psi, \psi \rangle_T \cdot^{\mathrm{D}} \mathbf{V} = \langle \mathbf{U}, \mathbf{V} \rangle$.
3. La norme canonique pour les tenseurs d'ordre deux est donnée par $\|\mathbf{U}\|_2 = \sqrt{\mathbf{U} \cdot^{\mathrm{D}} \mathbf{U}}$.
4. On le trouve également sous le nom de théorème d'Eckart-Young [Eckart et Young, 1936].

Définition 2.2. Soit un champ connu $u^{h,\Delta t}(x,t) = \phi(x) \otimes \psi(t) \overset{\mathrm{D}}{\cdot} \mathbf{U} \in \mathscr{U}_h^S \otimes \mathscr{U}_{\Delta t}^T$. Une meilleure approximation de ce champ dans $\mathscr{R}_M^{h,\Delta t}$, par rapport à la norme $\| \, \|_2$, est donnée par

$$u_M^{h,\Delta t}(x,t) = \sum_{m=1}^{M} \sigma_m w_m^h(x) \lambda_m^{\Delta t}(t) \quad \text{avec} \quad \begin{cases} w_m^h(x) & = \phi(x).\boldsymbol{W}_m \\ \lambda_m^{\Delta t}(t) & = \psi(t).\boldsymbol{\Lambda}_m \end{cases}, \tag{2.13}$$

où $\{\sigma_m, \boldsymbol{W}_m, \boldsymbol{\Lambda}_m\}_{m=1}^M$ sont les M premiers éléments singuliers de $\mathbf{U}$. Cette solution est unique si $\sigma_1 > \sigma_2 > \cdots > \sigma_M > 0$. De plus, pour $M = \mathrm{rang}(\mathbf{U})$, on a $u^{h,\Delta t} = u_M^{h,\Delta t}$.

Dans ce manuscrit, on utilise l'algorithme du logiciel *Matlab*, basé sur la librairie *LAPACK* [Anderson *et al.*, 1999], pour calculer la SVD de $\mathbf{U}$.

Remarque 2.6. *Le rang de* $\mathbf{U}$ *est défini par le nombre de ses valeurs singulières strictement positives.*

Remarque 2.7. *Les éléments singuliers* $\{\sigma_m, \boldsymbol{W}_m, \boldsymbol{\Lambda}_m\}_{m=1}^{rang(\mathbf{U})}$ *de* $\mathbf{U} \in \mathbb{R}^{n_S} \otimes \mathbb{R}^{n_T}$ *sont construits à partir des éléments propres de la matrice de corrélation spatiale* $\mathbf{U}.\mathbf{U}'$ *si* $n_S < n_T$ *et de la matrice de corrélation temporelle* $\mathbf{U}'.\mathbf{U}$ *si* $n_T < n_S$*. Notamment, chaque valeur singulière* σ_m *est associée à la racine carré d'une valeur propre de la matrice de corrélation. Les éléments singuliers sont ordonnés de la plus grande valeur singulière à la plus petite.*

Remarque 2.8. *Les modes en espace sont orthogonaux deux à deux par rapport au produit scalaire* $\langle , \rangle_S$ *et les modes en temps le sont par rapport au produit scalaire* $\langle , \rangle_T$*. De plus, dans le cas où les modes sont normalisés, on a la propriété suivante :*

$$\| u - u_M \|^2 = \| u \|^2 - \sum_{m=1}^{M} \sigma_m^2 \, . \tag{2.14}$$

Cas d'une norme pondérée

La construction de la meilleure approximation est similaire dans le cas où les matrices $\mathbf{N}_S$ et $\mathbf{N}_T$ sont différentes des matrices identités. Ces matrices étant symétriques définies positives, une méthode est de les décomposer sous la forme $\mathbf{N}_S = \mathbf{N}_S^{1/2}.\mathbf{N}_S^{1/2}$ et $\mathbf{N}_T = \mathbf{N}_T^{1/2}.\mathbf{N}_T^{1/2}$. La meilleure approximation de rang M est alors construite à partir de la SVD de $\left(\mathbf{N}_S^{1/2} \otimes \mathbf{N}_T^{1/2}\right) \overset{\mathrm{D}}{\cdot} \mathbf{U}$ (voir [Volkwein, 2008]). Dans ce manuscrit, on utilise une approche qui ne nécessite pas de décomposer les matrices $\mathbf{N}_S$ et $\mathbf{N_T}$. Le problème de minimisation (2.11) est résolu avec un algorithme glouton avec orthogonalisation des modes. La construction gloutonne d'une approximation de rang M est décrite dans le Chapitre 5.

Exemple 2.2. (Choix d'une norme) La meilleure approximation de rang M dépend de la norme utilisée dans la Définition 2.1. Dans cette exemple, on compare les approximations obtenues avec deux normes différentes, définies sur un espace de dimension finie. On note $\phi \otimes \psi$ une base éléments finis espace-temps de cet espace. On compare les normes suivantes :

$$\| u \|_{L2} = \Big(\int_I \int_\Omega u^2(x,t) \, \mathrm{dx} \, \mathrm{dt} \Big)^{1/2} \quad \text{et} \quad \| u \|_2 = \Big(\sum_{i=1}^{n_S} \sum_{j=1}^{n_T} u^2(x_i, t_j) \Big)^{1/2}, \tag{2.15}$$

où les points (x_i, t_j) sont les coordonnées des noeuds du maillage espace-temps, et $n_S = \dim(\mathscr{U}_h^S)$ et $n_T = \dim(\mathscr{U}_{\Delta t}^T)$. Ces deux normes permettent de minimiser l'erreur de façon équivalente en tout point du domaine espace-temps. Aussi, elles vérifient la propriété de séparabilité (2.10). Leurs représentations discrètes sont données par :

$$\| \mathbf{U} \|_{L2} = \big(\mathbf{U} \,\underline{\cdot}\, (\mathbf{N}_S \otimes \mathbf{N}_T) \,\underline{\cdot}\, \mathbf{U} \big)^{1/2} \quad \text{et} \quad \| \mathbf{U} \|_2 = \big(\mathbf{U} \,\underline{\cdot}\, (\mathbf{I}_S \otimes \mathbf{I}_T) \,\underline{\cdot}\, \mathbf{U} \big)^{1/2}, \tag{2.16}$$

avec $\mathbf{N}_S = \int_\Omega \phi \otimes \phi \, \mathrm{dx}$ et $\mathbf{N}_T = \int_\Omega \psi \otimes \psi \, \mathrm{dt}$, et $\mathbf{I}_S$ et $\mathbf{I}_T$ les matrices identités (on rappelle que la base éléments finis est interpolante en tout point du maillage espace-temps).

Pour comparer ces deux normes, on suppose que la champ connu que l'on cherche à approcher est la solution exacte (notée u) de l'équation des ondes, donnée dans l'Exemple 1.1. On projette tout d'abord ce champ sur $\mathscr{U}_h^S \otimes \mathscr{U}_{\Delta t}^T$. Le champ projeté est noté $\Pi(u)$ où $\Pi : \mathscr{U}^S \otimes \mathscr{U}^T \to \mathscr{U}_h^S \otimes \mathscr{U}_{\Delta t}^T$ est l'opérateur de projection orthogonale au sens du produit scalaire $\langle , \rangle_2$. On calcule ensuite la meilleure approximation de rang M,

$$u_M^{h,\Delta t} = \arg \min_{u^\star \in \mathscr{R}_M^{h,\Delta t}} \| \Pi(u) - u^\star \|_{2 \text{ ou } L2}, \tag{2.17}$$

dans les deux normes considérées. L'erreur (mesurée dans chacune des normes) entre $\Pi(u)$ et la meilleure approximation de rang M est présentée sur la Figure 2.2. Bien sûr, pour toutes les valeurs du rang M, la meilleure approximation au sens d'une des deux normes minimise l'erreur mesurée dans cette norme : la courbe rouge est en dessous de la courbe noire sur la figure de gauche et vice versa sur la figure de droite. Cependant, les courbes rouge et noire sont quasiment superposées sur les deux figures. Les approximations $u_M^{h,\Delta t}$ définies dans chacune des normes sont donc aussi proches de $\Pi(u)$, ceci que l'on mesure la distance dans l'une ou l'autre des normes.

Les approximations obtenues dans chacune des normes étant similaires, on choisit la méthode la moins coûteuse. **Le calcul de la meilleure approximation étant moins coûteux dans la norme $\| \ \|_2$, on utilise cette norme pour définir la meilleure approximation de rang M, dans tout le manuscrit.** Ce choix permet également de calculer une norme dans l'espace dual sans avoir à inverser de matrice (voir le Chapitre 6). Le choix de la norme $\| \ \|_{L2}$ nécessite en effet de calculer l'inverse de $\mathbf{N}_S$ et $\mathbf{N}_T$ pour calculer la norme associée dans l'espace dual. L'inconvénient de la norme $\| \ \|_2$ est qu'elle dépend

de la discrétisation (la norme de $\| \Pi(u) \|_2$ augmente lorsque l'on raffine le maillage). Aussi, on utilisera toujours une mesure relative pour comparer des distances calculées sur différents maillages.

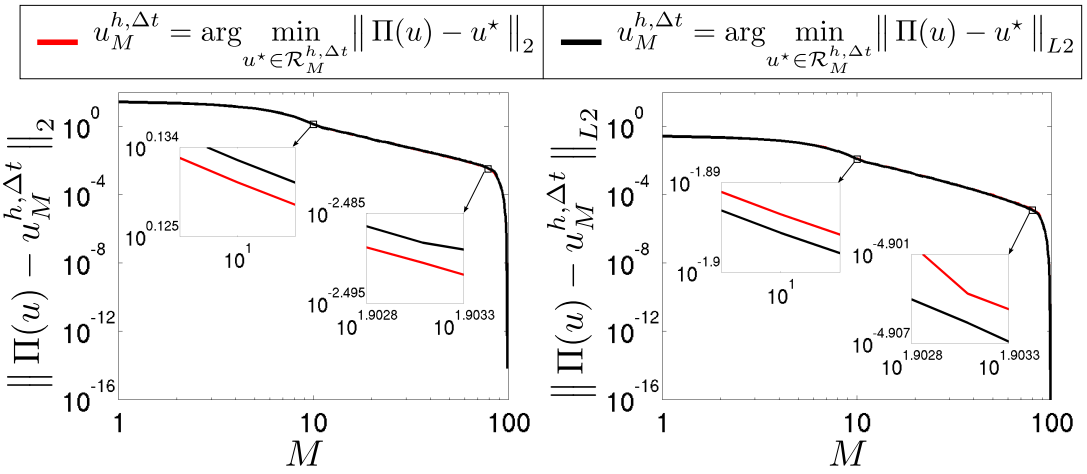

FIGURE 2.2: Comparaison des meilleures approximations de rang M définies dans deux normes différentes.

2.3 Efficacité en dynamique transitoire

Dans la section précédente, on a défini la meilleure approximation de rang M et décrit comment la construire. On peut maintenant évaluer si une approximation à variables séparées espace-temps est efficace dans le cas d'un problème de dynamique transitoire. On montre tout d'abord que le nombre de modes espace-temps, permettant d'approcher la solution d'un problème de choc à une erreur donnée, dépend du régime dynamique transitoire considéré. On introduit ensuite une démarche permettant d'évaluer un gain mémoire, qui sert de critère pour quantifier l'efficacité de la méthode de séparation de variables en terme de compression des résultats.

Remarque 2.9. *Dans toute cette section, l'efficacité de la méthode est illustrée avec le cas test unidimensionnel présenté dans l'Exemple 1.1. On rapelle qu'il s'agit d'une poutre* $\Omega = [0, L]$ *encastrée au point* $x = L$*, et soumise à un déplacement imposé de type choc au point* $x = 0$*. La durée caractéristique du choc est notée* ΔT*. On considère la réponse transitoire de la poutre en tout point de* Ω *sur l'intervalle de temps* $I = [0, T]$*. La réponse transitoire est caractérisée par le nombre sans dimension* $\kappa = (\frac{L}{c\Delta T})^2$ *où* c *est la célérité des ondes dans le milieu. Pour simplifier l'analyse, on prend* $c = 1\,m/s$ *et* $L = 1\,m$*.*

2.3.1 Description qualitative

On présente ici des résultats concernant la meilleure approximation de rang M de la solution exacte du problème de choc. Cette solution (notée u) est donnée dans $\mathscr{U}^S \otimes \mathscr{U}^T$ par l'équation (1.8). La construction de sa meilleure approximation (notée u_M) dans $\mathscr{R}_M$ est décrite dans l'Exemple 2.3. L'erreur relative entre la solution exacte et sa meilleure approximation de rang M est donnée par :

$$\mathrm{err}^{\text{rom-exact}}(M) = \frac{\left\| u - u_M \right\|_2}{\left\| u \right\|_2}. \tag{2.18}$$

On notera que cette erreur est uniquement due au rang de l'approximation.

Influence du régime dynamique transitoire

On décrit tout d'abord l'influence du régime dynamique transitoire sur l'efficacité de la méthode de séparation de variables. Pour cela, on calcule la meilleure approximation de rang M de la solution exacte, pour différentes valeurs du nombre κ. Les résultats sont présentés sur la Figure 4.1. On peut faire les commentaires suivants :

- L'erreur due à une approximation de rang M pour les différentes valeurs de κ peut être approchée par la fonction $\epsilon(M,\kappa)$ définie par :

$$\boxed{\epsilon(M,\kappa) = C\left(\frac{\kappa^a}{M}\right)^b}, \tag{2.19}$$

  où les constantes C et b sont indépendantes de M et κ, et la constante a dépend uniquement de κ. On obtient les valeurs suivantes : $C = 1.1$, $b = 2.64$, et $a = 0.37$ pour $\kappa < 1$ ou $a = 0.48$ pour $\kappa \geq 1$ (voir la comparaison sur la Figure 2.4). Cette formule peut être utilisée [5] pour déterminer le nombre de modes nécessaires pour obtenir une approximation d'une précision donnée en fonction du nombre κ.

- Pour les valeurs de $\kappa < 1$, l'erreur due à une approximation de rang un, tend vers zéro lorsque que κ diminue (voir les points $M = 1$ sur la Figure 4.1). Ceci illustre le fait que la solution du problème dynamique tend vers la solution du problème quasi-statique lorsque κ tend vers zéro. On verra en effet dans la Remarque 4.2, que la solution du problème quasi-statique admet une représentation exacte en variables séparées espace-temps de rang un [6]. Pour les faibles valeurs de κ, seulement quelques modes espace-temps sont donc nécessaires pour approcher la solution avec une très bonne précision.

5. Bien sûr, la formule (2.19) n'est valable que pour le cas test considéré. On notera cependant que la même allure des courbes (de convergence de l'erreur en fonction du rang et du nombre κ) est observée pour un cas test avec condition aux limites de Neumann.

6. Cette remarque est valable lorsque le chargement peut se mettre sous la forme $f(x,t) = f^S(x)f^T(t)$ (« chargement radial »), ce qui est le cas ici si on néglige le terme d'accélération dans l'équation d'équilibre.

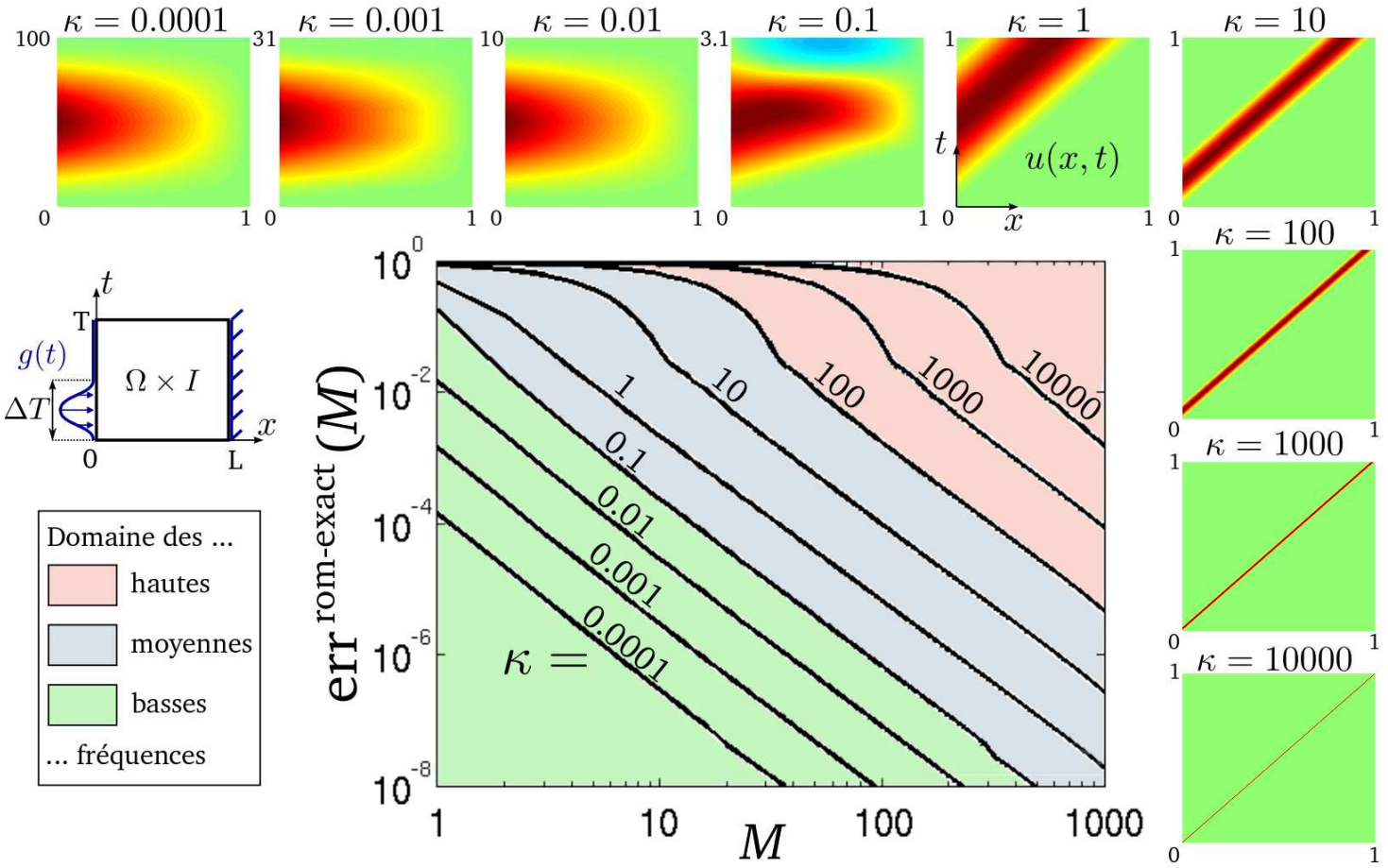

FIGURE 2.3: Erreur relative entre la solution exacte $u(x,t)$ et sa meilleure approximation de rang M pour différentes valeurs du nombre κ.

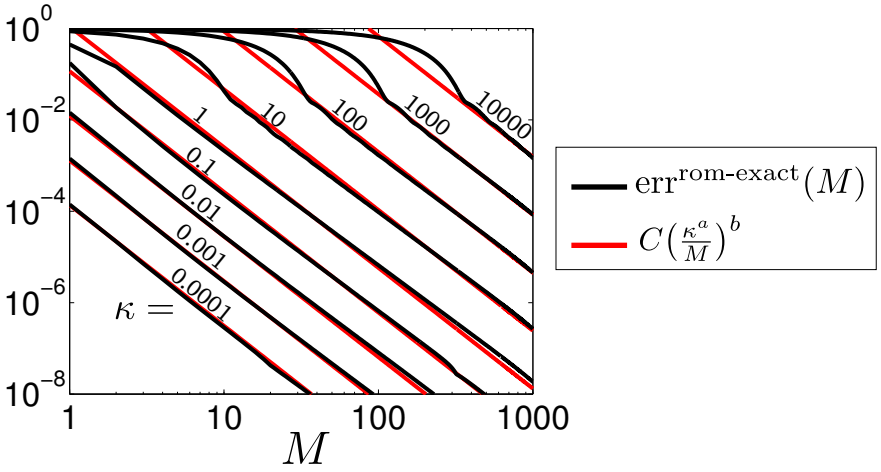

FIGURE 2.4: Comparaison entre les courbes de la Figure 4.1 et la fonction $\epsilon = C\left(\frac{\kappa^a}{M}\right)^b$.

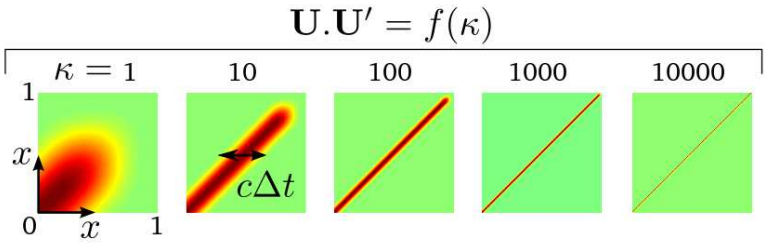

FIGURE 2.5: Matrice de corrélation spatiale pour différentes valeurs de κ.

- Pour les valeurs de $\kappa \geq 1$, les premiers modes espace-temps ne diminuent pas l'erreur due à l'approximation de rang M. Puis, à partir d'une certaine valeur M_0, on observe une convergence linéaire (en échelles logarithmiques) de l'erreur d'approximation en fonction du rang M, dont la pente ne dépend pas de κ. En revanche, le rang minimum M_0 (à partir duquel l'erreur diminue de façon significative) dépend du nombre κ. Avec la formule (2.19), on montre que cette valeur de M_0 est proportionnelle à $\kappa^a \simeq \sqrt{\kappa}$. Techniquement, la convergence de l'erreur en fonction du rang peut s'expliquer en analysant la matrice de corrélation $\mathbf{U}.\mathbf{U}'$ pour les différentes valeurs de κ. Cette matrice tend vers une matrice « bande » dont la largeur est de plus en plus faible lorsque κ tend vers l'infini. Aussi, la diagonale de cette matrice est caractérisée par des valeurs constantes presque partout, sauf dans la « longueur $c\Delta T$ » caractérisant le support spatial de la perturbation provoquée par le choc (voir la Figure 2.5). Ainsi, lorsque κ tend vers l'infini, les valeurs propres de la matrice de corrélation tendent vers une unique valeur propre de multiplicité infinie. On comprend alors que l'erreur due à l'approximation de rang M diminue très lentement dans ce cas (voir l'équation (2.14)).
- On observe clairement une zone de transition pour les valeurs de $\kappa \in [0.1, 100]$, qui correspond au régime moyenne fréquence (voir la zone bleutée sur la Figure

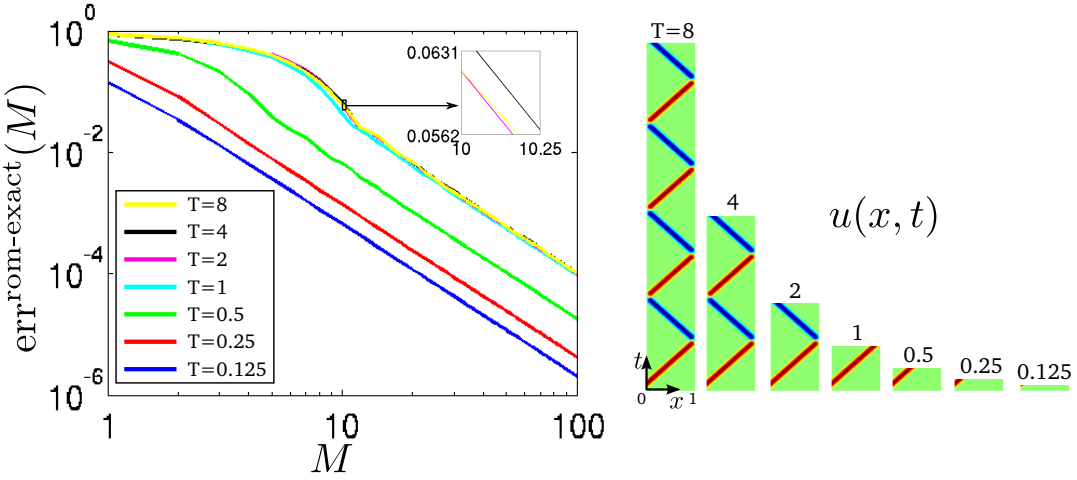

FIGURE 2.6: Influence de la durée T de la simulation sur l'erreur entre la solution exacte et son approximation de rang M (avec $\kappa = \mathrm{cst} = 10$).

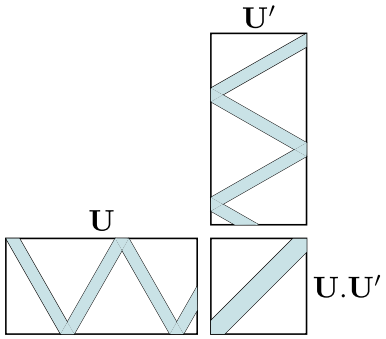

FIGURE 2.7: Construction de la matrice de corrélation spatiale.

4.1). Pour ce régime dynamique, l'efficacité de la méthode de séparation de variables espace-temps fait débat. Aussi, on se concentre sur cette zone dans la suite du manuscrit.

Influence de la durée de la simulation

Intuitivement, on pourrait penser que l'erreur entre la solution exacte et son approximation de rang M dépend du nombre de fois que l'onde parcourt le milieu. Pour évaluer cet aspect, on représente sur la Figure 2.6 l'erreur due à une approximation de rang M obtenue, à κ constant, pour différentes valeurs de la durée T de la simulation. Pour des durées inférieures à $T = 1$, l'erreur obtenue pour un nombre de modes M constant, diminue avec la durée de la simulation. Dans ce cas, la simulation est trop courte pour que l'onde atteigne l'autre extrémité de la poutre, et la portion $[cT, L]$ de la poutre n'est pas sollicitée. Le cas test est ainsi équivalent au cas où la longueur de la poutre est fixée à cT, ce qui revient à diminuer la valeur de κ, et explique donc que l'erreur d'approximation à M constant diminue avec la durée de simulation T. À l'op-

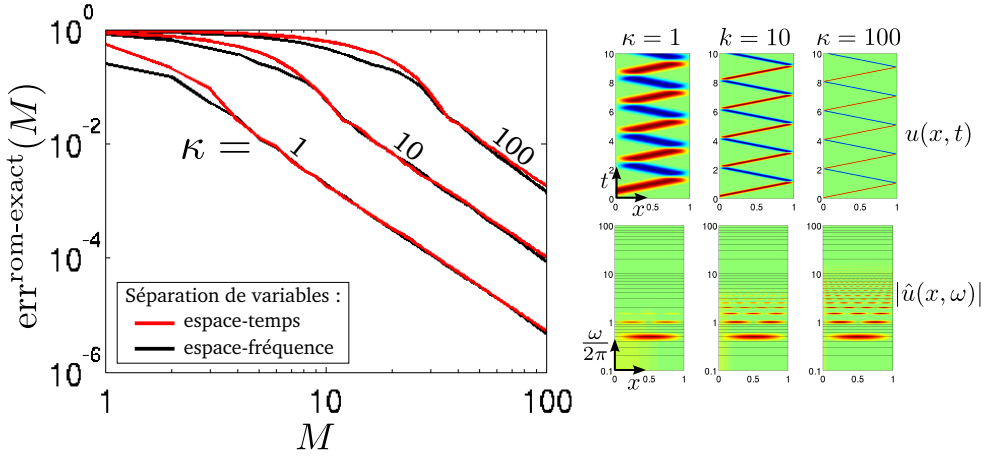

FIGURE 2.8: Erreurs dues à une approximation de rang M pour les méthodes de séparations de variables espace-temps et espace-fréquence.

posé, on observe qu'à partir de $T = 1$, qui correspond à l'arrivée de l'onde en bout de poutre, l'erreur due à une approximation de rang M est très semblable quelque soit la durée T de la simulation. Techniquement, on observe que les parcours successifs de l'onde dans le domaine spatiale ne modifie pas la structure de la matrice de corrélation spatiale $\mathbf{U}.\mathbf{U}'$ (voir l'illustration de sa construction sur la Figure 2.7). À κ constant, la méthode de séparation de variables sera donc d'autant plus efficace que la simulation durera longtemps.

Séparations de variables espace-temps & espace-fréquence

Dans ce manuscrit, le problème de dynamique est modélisé sur le domaine espace-temps. Aussi, il est intéressant d'évaluer l'efficacité de la méthode de séparation de variables dans le cas d'une modélisation en variables espace-fréquence. Pour cela, on calcule la meilleure approximation de rang M (en variables séparées espace-fréquence) de la transformée de Fourier $\hat{u}(x,\omega)$ de la solution exacte, introduite dans l'Exemple 1.1. Le champ $\hat{u}(x,\omega)$ est un champ complexe. Aussi, on calcule la meilleure approximation de rang M de la partie réelle de ce champ. La même allure des courbes est obtenue pour la partie imaginaire. Les résultats obtenus pour $\kappa = 1 - 10 - 100$ sont présentés sur la Figure 2.8. On observe que le rang nécessaire pour obtenir une erreur d'approximation donnée, est très similaire que l'on considère une séparation espace-temps ou espace-fréquence. L'efficacité d'une approche par rapport à l'autre dépend alors du choix des espaces d'approximation utilisés pour approcher les modes espace-temps ou espace-fréquence. Dans la section suivante, on étudie l'influence de ce choix dans le cas d'une modélisation en variables espace-temps.

Exemple 2.3. (Construction de la meilleure approximation dans $\mathscr{R}_M$) Dans cet exemple, on décrit la construction (de l'approximation de) de la meilleure approximation de rang M d'une fonction u appartenant à un espace de dimension infinie, noté $\mathscr{U}^S \otimes \mathscr{U}^T$. La meilleure approximation de cette fonction dans $\mathscr{R}_M$ est notée u_M. L'idée est simplement d'approcher u_M dans un espace de dimension finie que l'on note $\mathscr{U}_l^S \otimes \mathscr{U}_l^T$. Le sous-ensemble de $\mathscr{U}_l^S \otimes \mathscr{U}_l^T$ contenant les représentations à variables séparées espace-temps de rang M est noté $\mathscr{R}_{M,l}$. L'espace d'approximation d'indice l est construit par la méthode des éléments finis avec un pas de discrétisation égale à $\frac{h}{2^l}$ en espace et $\frac{\Delta t}{2^l}$ en temps. La démarche est schématisée sur la Figure 2.9. On répète les étapes suivantes en commençant à la grille $l = 0$:

1. on calcule $\Pi(u)$, la projection de u sur l'espace $\mathscr{U}_l^S \otimes \mathscr{U}_l^T$,
2. on calcule $u_{M,l}$, la meilleure approximation de rang M de $\Pi(u)$ dans $\mathscr{R}_{M,l}$,
3. on calcule $d_l(M)$, l'erreur relative définie par $d_l(M) = \left\| \Pi(u) - u_{M,l} \right\|_2 / \left\| \Pi(u) \right\|_2$,
4. si $\frac{|d_l(M) - d_{l-1}(M)|}{d_l(M)} <$ *tolérance* alors on considère que $u_{M,l} \in \mathscr{R}_{M,l}$ est une bonne approximation de $u_M \in \mathscr{R}_M$. Sinon on passe à la grille $l = l + 1$.

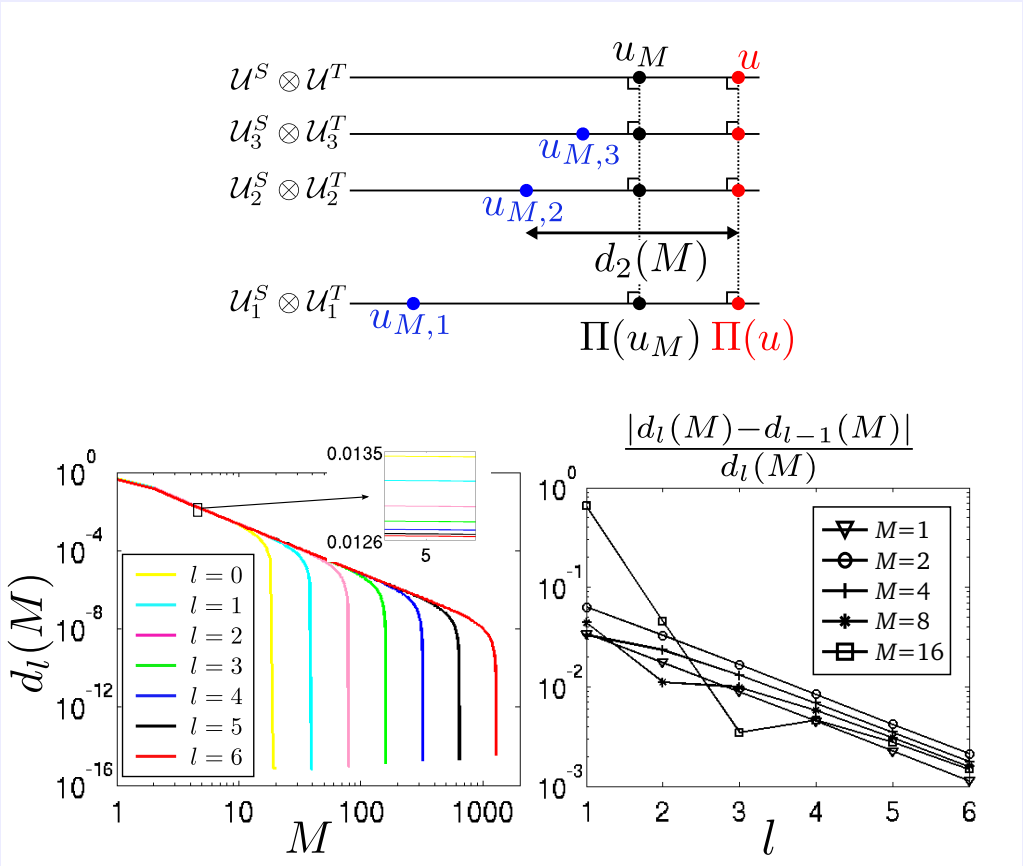

FIGURE 2.9: Convergence de $u_{M,l}$ vers $u_M \in \mathscr{R}_M \subset \mathscr{U}^S \otimes \mathscr{U}^T$ lorsque l augmente.

2.3.2 Description quantitative

Dans le cas général, la solution exacte du problème considéré n'est pas connue. On connait seulement une approximation de celle-ci, calculée dans un espace de dimension finie. C'est la mémoire nécessaire au stockage de cette solution discrète que l'on souhaite réduire à l'aide d'une représentation à variables séparées. Aussi, pour évaluer l'efficacité de la méthode, on doit prendre en compte (et comparer) deux sources d'erreur, liées d'une part à la discrétisation et d'autre part au rang M de l'approximation à variables séparées. On introduit ici un critère prenant en compte ces deux sources d'erreur pour déterminer le gain mémoire relatif au stockage des résultats sous format séparé.

Différentes sources d'erreurs

Pour expliciter ces deux sources d'erreur, on note $u \in \mathscr{U}^S \otimes \mathscr{U}^T$ la solution exacte du problème considéré, $u^{h,\Delta t} \in \mathscr{U}_h^S \otimes \mathscr{U}_{\Delta t}^T$ l'approximation de u, qui est connue et que l'on souhaite compresser, et $u_M^{h,\Delta t} \in \mathscr{R}_M^{h,\Delta t}$ la meilleure approximation de rang M de $u^{h,\Delta t}$, que l'on appelle la solution compressée. L'erreur entre la solution exacte et la solution compressée peut être décomposée comme suit [7] :

$$\boxed{\underbrace{\left\| \Pi(u) - u_M^{h,\Delta t} \right\|_2}_{\text{erreur totale}} \leq \underbrace{\left\| \Pi(u) - u^{h,\Delta t} \right\|_2}_{\text{erreur de discrétisation}} + \underbrace{\left\| u^{h,\Delta t} - u_M^{h,\Delta t} \right\|_2}_{\text{erreur de décomposition}}}. \tag{2.20}$$

On définit les erreurs relatives suivantes qui sont schématisées sur la Figure 2.10 :

$$\mathrm{err}^{\mathrm{tot}}(M,h,\Delta t) = \frac{\left\| \Pi(u) - u_M^{h,\Delta t} \right\|_2}{\left\| \Pi(u) \right\|_2} \quad \text{(erreur totale)}, \tag{2.21a}$$

$$\mathrm{err}^{\mathrm{disc}}(h,\Delta t) = \frac{\left\| \Pi(u) - u^{h,\Delta t} \right\|_2}{\left\| \Pi(u) \right\|_2} \quad \text{(erreur de discrétisation)}, \tag{2.21b}$$

$$\mathrm{err}^{\mathrm{rom}}(M,h,\Delta t) = \frac{\left\| u^{h,\Delta t} - u_M^{h,\Delta t} \right\|_2}{\left\| \Pi(u) \right\|_2} \quad \text{(erreur de décomposition)}. \tag{2.21c}$$

L'évolution de ces erreurs en fonction du rang M et de la dimension de l'espace d'approximation est représentés sur la Figure 2.11. L'espace d'approximation $\mathscr{U}_h^S \otimes \mathscr{U}_{\Delta t}^T$ est construit avec la méthode des éléments finis P1 en espace et différentes méthodes d'approximation en temps : le schéma de Newmark (avec $\beta = 1/4$ et $\gamma = 1/2$), la méthode de Galerkin discontinue en temps à un champ (TDG P2) et deux champs (TDG P1-P1), et la méthode de Galerkin continue en temps à deux champs (TG P1-P1)). Le cas test est toujours celui de l'Exemple 1.1 avec $\kappa = 10$ et $T = 1$s.

7. On rappelle que $\Pi(u)$ est la projection orthogonale de $u \in \mathscr{U}^S \otimes \mathscr{U}^T$ sur l'espace $\mathscr{U}_h^S \otimes \mathscr{U}_{\Delta t}^T$.

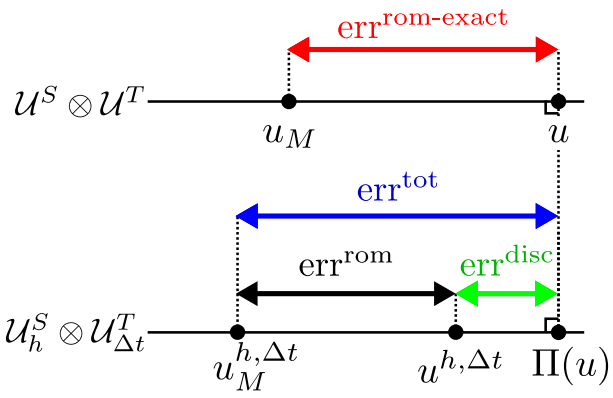

FIGURE 2.10: Décomposition de l'erreur totale en une partie due à la discrétisation et une partie due à l'approximation de rang M.

On observe que l'erreur totale (courbes bleues) diminue lorsque le rang M augmente. Puis, à partir d'une certaine valeur du rang, l'erreur totale stagne au niveau de l'erreur de discrétisation (courbes vertes). Il est alors inutile de calculer d'autres modes espace-temps puisqu'à partir de cette valeur du rang, l'erreur de décomposition (courbes noires) devient négligeable devant l'erreur de discrétisation. Sur cette figure, on représente également l'erreur entre la solution exacte u et son approximation u_M dans $\mathscr{R}_M$ (courbe rouge). On rappelle que cette erreur dépend uniquement du rang M de l'approximation (et pas de la discrétisation). Elle est définie par :

$$\mathrm{err}^{\text{rom-exact}}(M) = \frac{\left\| u - u_M \right\|_2}{\left\| u \right\|_2} \qquad \text{(erreur exacte de décomposition)}. \tag{2.22}$$

Quelle que soit la méthode d'approximation choisie, l'erreur $\mathrm{err}^{\text{tot}}(M,h,\Delta t)$ converge vers l'erreur $\mathrm{err}^{\text{rom-exact}}(M)$ lorsque les paramètres h et Δt tendent vers zéro. Ainsi, pour une erreur de discrétisation suffisamment faible, l'erreur totale dépend uniquement du rang M (et pas du choix de la méthode d'approximation).

Gain mémoire

Pour un espace d'approximation donné, il est inutile de choisir un rang M tel que l'erreur de décomposition soit très faible devant l'erreur de discrétisation (et vice-versa). On propose alors la démarche suivante pour évaluer l'efficacité de la méthode de séparation de variables en terme de compression de données. On cherche tout d'abord les paramètres de discrétisation h et Δt tels que l'erreur $\mathrm{err}^{\text{disc}}(h,\Delta t)$ soit égale à une certaine valeur $\epsilon \pm \epsilon^{tol}$ (voir l'étape 1 sur la Figure 2.12). Puis on cherche le rang minimum $M_{\min}$ tel que l'erreur $\mathrm{err}^{\text{rom}}(M_{\min},h,\Delta t)$ soit inférieure à ϵ (voir l'étape 2 sur la Figure 2.12). La solution compressée obtenue de cette façon est aussi précise [8] que la solution discrète par rapport à la solution exacte. On définit alors le « gain mémoire »

8. On a dans ce cas $d^{tot} \leq 2\epsilon \pm \epsilon^{tol}$.

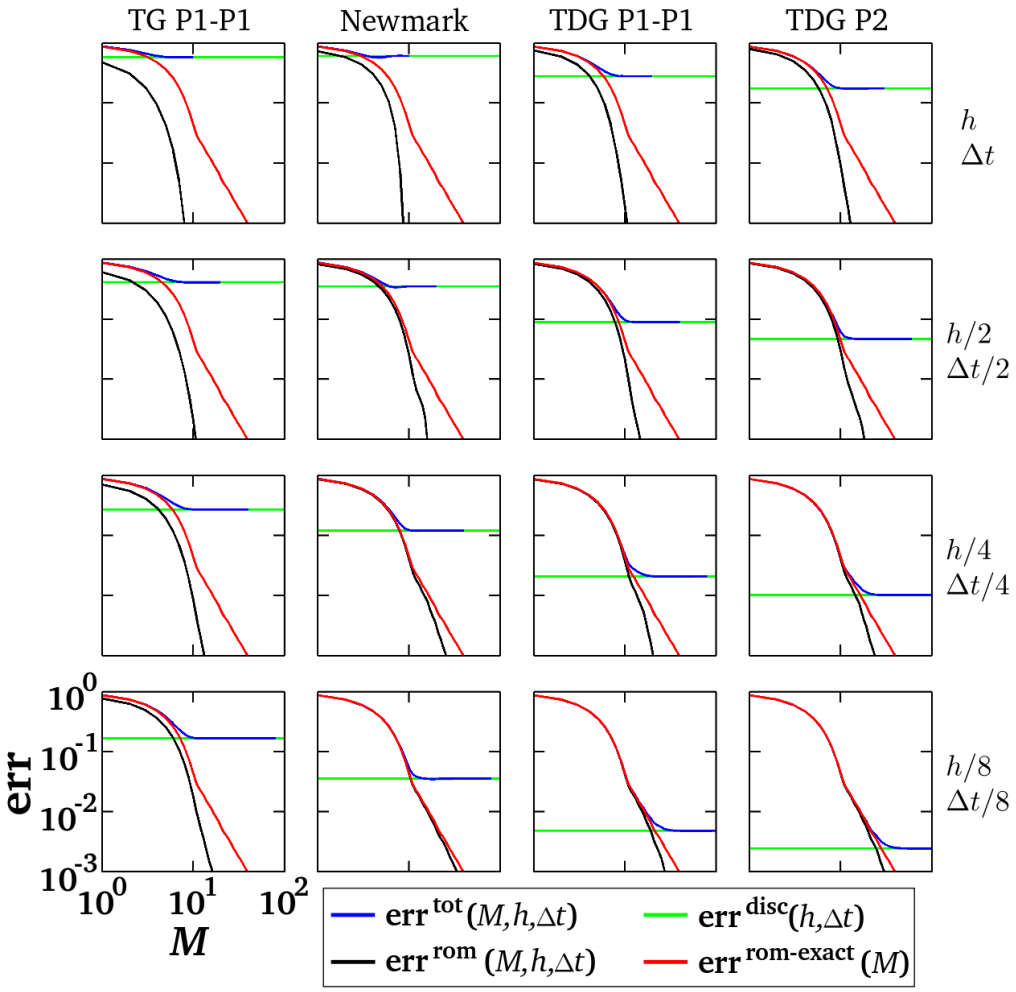

FIGURE 2.11: Évolution des différentes sources d'erreur en fonction du rang M, des paramètres de discrétisation h et Δt, et de la méthode d'approximation en temps.

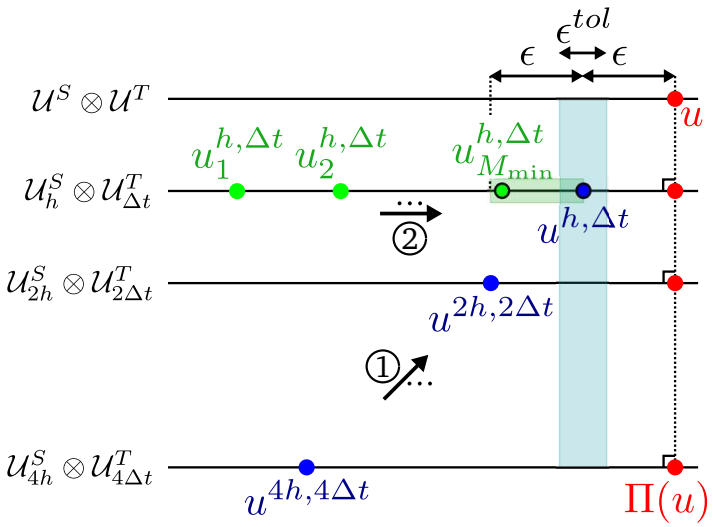

FIGURE 2.12: Démarche suivie pour évaluer le gain mémoire associé à la méthode de séparation de variables.

comme le rapport entre la mémoire nécessaire au stockage de la solution discrète et la mémoire nécessaire au stockage de la solution compressée. Si ce gain mémoire est supérieur à un, alors la méthode de séparation de variables est un outil efficace pour compresser les résultats de la simulation considérée.

Définition 2.3. Soit une approximation de la solution d'un problème donné, appartenant à un espace de dimension finie $\mathcal{U}_h^S \otimes \mathcal{U}_{\Delta t}^T$ et telle que l'erreur entre cette solution discrète et la solution exacte soit égale à ϵ. Alors, le gain mémoire apporté par la méthode de séparation de variables est défini par

$$\text{gain mémoire} = \frac{n_S \times n_T}{M_{\min} \times (n_S + n_T)}, \tag{2.23}$$

où $n_S = \dim(\mathcal{U}_h^S)$ et $n_T = \dim(\mathcal{U}_{\Delta t}^T)$, et $M_{\min}$ est le rang minimum tel que l'erreur entre la solution discrète et sa meilleure approximation dans $\mathcal{R}_M^{h,\Delta t}$ soit inférieure à ϵ.

Gain mémoire pour différentes méthodes d'approximation

On compare ici le gain mémoire obtenu avec différentes méthodes d'approximation espace-temps, à savoir :

- la méthode des éléments finis P1 en espace avec l'approximation de Newmark en temps ($\beta = 1/4$ et $\gamma = 1/2$),

- les méthodes des éléments finis espace-temps (discontinue en temps) à un champ (TDG U) et deux champs (TDG UV) avec des polynômes de Lagrange de degré p en espace et en temps.

On utilise le cas test de l'Exemple 1.1 avec $\kappa = 1 - 10 - 100$ et $T = 10$s. Le gain mémoire est calculé pour une erreur relative de discrétisation égale à $0.1 \pm 0.05\%$ pour toutes les valeurs de κ et pour toutes les méthodes d'approximation. Les résultats sont présentés dans le Tableau 2.1. On peut faire les remarques suivantes :

- À précision équivalente, la dimension de l'espace d'approximation est plus grande avec le schéma de Newmark qu'avec les méthodes éléments finis espace-temps, tranduisant le fait que le schéma de Newmark est moins précis que les méthodes élements finis espace-temps. La différence entre les deux classes de méthodes est accentuée lorsque l'on augmente la valeur de κ. Par exemple, pour $\kappa = 100$, la dimension $n_S n_T$ de l'espace d'approximation est 288 fois plus grande avec le schéma de Newmark qu'avec la méthode TDG UV de degré $p = 5$.
- Pour toutes les valeurs de κ, approximativement le même nombre de modes espace-temps M_{min} est obtenu pour les différentes méthodes d'approximation. Dans tous les cas considérés, le rang M_{min} est inférieur au rang théorique [9] donné par la formule (2.19).
- Comme le rang M_{min} dépend très peu de la méthode d'approximation choisie, le gain mémoire est logiquement le plus élevé pour la méthode d'approximation la moins précise. Dans tous les cas, on observe un gain supérieur à 4. Pour le schéma de Newmark, on observe un gain de plus d'un ordre de grandeur. Pour $\kappa = 100$, il faut par exemple 58 fois moins de mémoire pour stocker la solution compressée que pour stocker la solution discrète, à précision équivalente. De plus, le gain mémoire est plus important pour $\kappa = 1$ que pour $\kappa = 100$. Ceci s'explique par le fait que le rang M_{min} augmente de façon moins importante que la dimension de l'espace d'approximation, lorsque κ augmente. On observe en effet que M_{min} augmente de façon proportionnelle à $\kappa^{0.5}$, alors que $n_S n_T$ augmente proportionnellement à $\kappa^{1.28}$.
- Finalement, même si le gain mémoire est plus important pour les méthodes d'approximation les moins précises, la mémoire compressée reste moins grande pour les méthodes d'approximation les plus précises. Par exemple, pour $\kappa = 100$, il faut 4 Mo pour stocker la solution compressée avec la méthode TDG P5-P5 alors que le stockage de la solution compressée avec le schéma de Newmark nécessite 82 Mo, soit 20 fois plus de mémoire.

9. La formule (2.19) donne le rang M en fonction de l'erreur $\text{err}^{\text{rom-exact}}$ et du nombre κ. Pour $\text{err}^{\text{rom-exact}} = 1e-3$, on obtient $M = 14 - 43 - 129$ pour $\kappa = 1 - 10 - 100$ respectivement.

$\kappa \rightarrow$		1	10	100	1	10	100	1	10	100	1	10	100	1	10	100
Schéma	p	$n_S/10$			$n_T/100$			$M_{\min}$			Gain mémoire			Mémoire compressée (Mo)		
TDG-UV	2	**8**	24	97	6	26	131	**11**	35	112	**6.8**	6.3	8.1	5.9e-02	7.7e-01	1.2e+01
	3	13	23	61	5	19	73	12	36	108	8.6	5.6	5.2	6.0e-02	5.9e-01	6.5e+00
	4	12	**17**	56	5	15	56	12	35	107	8.1	4.4	4.7	5.3e-02	4.4e-01	5.0e+00
	5	12	**17**	**48**	**4**	**12**	**45**	**11**	34	105	8.4	**4.3**	**4.2**	**4.1e-02**	**3.4e-01**	**4.0e+00**
TDG-U	2	24	69	289	18	78	393	12	35	107	17.7	18.1	25.1	1.9e-01	2.3e+00	3.4e+01
	3	20	35	94	8	29	116	12	34	103	13.3	9.3	8.4	9.0e-02	8.5e-01	9.9e+00
	4	17	21	70	6	18	70	12	**33**	**102**	11.0	5.7	6.2	6.8e-02	5.0e-01	6.0e+00
	5	15	26	60	5	18	54	12	36	**102**	9.7	6.3	5.3	5.7e-02	5.5e-01	4.7e+00
Newmark	1	**50**	**167**	**806**	**34**	**150**	**772**	**13**	**42**	**126**	**33.9**	**35.7**	**57.9**	**3.9e-01**	**5.3e+00**	**8.2e+01**

TABLE 2.1: Influence du choix du l'espace d'approximation sur l'efficacité de la méthode de séparation de variables.
Les colonnes n_S et n_T sont les dimensions spatiale et temporelle de l'espace d'approximation, $M_{\min}$ est le rang minimum décrit dans la Définition 2.3 du Gain mémoire, et la Mémoire compressée est donnée par $M_{min}(n_S + n_T)$ en mégaoctets (Mo).

Exemple 2.4. (Complexité & Résolution incrémentale) Dans cet exemple, on compare la complexité associée à une résolution incrémentale du problème espace-temps, avec les différentes méthodes d'approximation utilisées dans le Tableau 2.1.

Comme on l'a vu au Chapitre 1, la complexité d'un schéma incrémental implicite est donnée par $N_T \mathbf{lin}(n)$ où N_T est le nombre d'intervalles de temps et n la taille du système linéaire que l'on doit résoudre à chaque incrément de temps. Ce système linéaire dépend de la méthode considérée (de part sa taille, sa population ou encore son conditionnement). Aussi, pour comparer les différentes méthodes, il est nécessaire de pouvoir évaluer la complexité associée à la résolution d'un système linéaire.

Une telle évaluation dépend d'une part du solveur de système linéaire utilisé et d'autre part du problème considéré. On choisit ici d'évaluer simplement la complexité d'une itération d'un solveur itératif préconditionné, en suivant la démarche proposée par [Huerta *et al.*, 2013]. Cette évaluation prend en compte le calcul d'un produit matrice-vecteur creux et le préconditionnement du système par une factorisation incomplète de type LU. La complexité de ces opérations est donnée par

$$G_{it} = 4\mathrm{nnz} + \mathrm{ndof}, \tag{2.24}$$

où nnz est le nombre d'entrées différentes de zéros dans la matrice et ndof le nombre d'équations du système linéaire. On suppose alors que la résolution nécessite autant d'itérations pour les différentes méthodes comparées. De cette façon, le ratio $G_{it}^{\text{méthode 1}} / G_{it}^{\text{méthode 2}}$ donne un critère de comparaison entre les méthodes.

On compare ici le schéma de Newmark avec éléments finis de degré p_S en espace et les méthodes TDG U et TDG UV avec éléments finis de degré p_S en espace et p_T en temps. Les valeurs de nnz et ndof, associées à la matrice du système linéaire que l'on doit résoudre à chaque incrément de temps, sont présentées dans le Tableau 2.2 pour ces différentes méthodes. On note N_S le nombre d'éléments du maillage spatial et on néglige la substitution associée aux conditions de Dirichet.

Méthode	nnz	ndof
Newmark	$(p_S^2 + 2p_S)N_S + 1$	$p_S N_S + 1$
TDG U	$(p_T+1)^2((p_S^2+2p_S)N_S+1)$	$(p_T+1)(p_S N_S+1)$
TDG UV	$4(p_T+1)^2((p_S^2+2p_S)N_S+1)$	$2(p_T+1)(p_S N_S+1)$

TABLE 2.2: Caractéristique du problème spatial à chaque incrément de temps.

On utilise alors les valeurs de N_S et N_T que l'on a identifiées dans le Tableau 2.1 pour calculer $N_T G_{it}$. On rappelle que ces paramètres du maillage espace-temps sont tels que l'erreur de discrétisation est identique pour toutes les méthodes. En

comparant alors la valeur de $N_T G_{it}$ pour les différentes méthodes, on peut estimer (grossièrement) quelle est la méthode la plus coûteuse en terme de complexité. Les ratios obtenus en prenant le schéma de Newmark comme référence sont reportés dans le Tableau 2.3.

Méthode	$p_S = p_T = p$	N_S	N_T	$(N_T G_{it})^{\text{Newmark}} / (N_T G_{it})^{\text{Méthode}}$
Newmark	1	8065	77185	1
TDG UV	2	485	4373	3.3
	3	205	1825	5.6
	4	139	1111	5.4
	5	97	745	5.5
TDG U	2	1444	13108	1.5
	3	313	2905	9.1
	4	175	1393	13.8
	5	121	898	14.7

TABLE 2.3: Complexité associée à une résolution incrémentale.

Les résultats obtenus montrent que les méthodes de Galerkin discontinues en temps sont plus rapides que le schéma de Newmark. On notera que le meilleur ratio est obtenu avec la méthode TDG P5 et que la méthode TDG UV est pénalisée par le fait que l'on ait à résoudre un problème à deux champs. Cependant, l'estimation effectuée ici est assez grossière et une mesure expérimentale du temps de calcul permettrait de confirmer le ratio estimé. On suppose notamment que le solveur itératif converge avec le même nombre d'itérations pour les différentes méthodes. Ceci sous entend que le conditionnement du système linéaire est similaire et avantage clairement les méthodes de Galerkin discontinues de degré élevé. Pour ce cas test, le conditionnement (mesuré avec Matlab) du système linéaire à résoudre à chaque incrément de temps est en effet inférieure à 10 pour le schéma de Newmark et de l'ordre de 10^7 pour les méthodes TDG de degré 5 en espace et en temps.

2.4 Conclusion

Dans ce chapitre, on a évalué l'intérêt de la méthode de séparation de variables à réduire la mémoire nécessaire au stockage des résultats d'une simulation de dynamique transitoire sur le domaine espace-temps. On a tout d'abord défini la meilleure approximation de rang M d'un champ connu, au sens d'un problème de minimisation dans

une certaine norme. La construction de cette meilleure approximation à l'aide de la décomposition en valeur singulière (SVD) a été précisée.

En pratique, le champ que l'on cherche à compresser est la solution discrète du problème de référence. Cette solution discrète est une approximation de la solution exacte de ce problème. Aussi, pour évaluer l'efficacité de la méthode de séparation de variables, le point clé est de comparer l'erreur entre la solution exacte et la solution discrète (appelée l'erreur de discrétisation) avec l'erreur entre la solution discrète et sa meilleure approximation de rang M (appelée l'erreur de décomposition). Pour un espace d'approximation de dimension donnée, il est en effet inutile de choisir un rang M tel que l'erreur de décomposition soit très faible devant l'erreur de discrétisation (et vice-versa).

Un critère basé sur l'identification du rang minimum M_{min} tel que la solution compressée soit aussi précise que la solution discrète, a été proposé pour comparer le gain mémoire obtenu pour différents régimes dynamiques transitoires et différentes méthodes d'approximation. **Au vue des gains obtenus (réduction de la mémoire d'un facteur variant de 4 à 58), on peut conclure que la méthode de séparation de variables permet de réduire efficacement la mémoire nécessaire au stockage des résultats** du problème de dynamique transitoire considéré.

Jusqu'ici on a supposé que le champ que l'on cherche à décomposer est connu. On a donc calculé puis stocké ce champ avant de le compresser. Cette démarche nécessite d'une part d'avoir le temps de calculer ce champ, et d'autre part de disposer de suffisamment de mémoire vive pour le stocker. Aussi, dans la suite du manuscrit, l'objectif est de calculer la meilleure approximation de rang M (ou tout du moins une bonne approximation de celle-ci) sans avoir à calculer explicitement le champ que l'on cherche à décomposer, ni à stocker un champ sous format non séparé. La démarche proposée pour atteindre cet objectif est basée sur les méthodes de réduction de modèle.

Les méthodes de réduction de modèle classiquement utilisées en dynamique transitoire, sont basées sur la projection des équations de mouvement sur une base de fonctions spatiales, de dimension réduite par rapport à la dimension de l'espace d'approximation. Ces méthodes sont présentées dans le chapitre suivant et leur performance est évaluée dans le cas d'un problème académique de choc. On compare notamment l'approximation à variables séparées obtenue avec ces méthodes, à la meilleure approximation de rang M introduite dans ce chapitre.

Chapitre 3

Méthodes de réduction de modèle par projection sur une base réduite

Dans ce chapitre, on compare l'efficacité des méthodes de réduction de modèle classiquement utilisées en dynamique transitoire, avec la meilleure approximation de rang M introduite dans le chapitre précédent.

Sommaire

3.1 Introduction

Dans le chapitre précédent, on a supposé que le champ que l'on cherche à décomposer est connue. On a donc calculé puis stocké celui-ci sur tout le domaine espace-temps avant de le compresser. En pratique, le calcul et le stockage de ce champ sont trop coûteux pour être réalisés. Aussi, l'objectif des méthodes de réduction de modèle est de calculer une bonne approximation de ce champ à moindre coût. Ces méthodes aboutissent également à une approximation de la solution de référence sous la forme d'une représentation à variables séparées espace-temps. On peut alors s'interroger sur l'optimalité de cette approximation à variables séparées par rapport à la meilleure approximation de rang M introduite précédemment. On établit une telle comparaison dans ce chapitre.

En dynamique des structures, le concept de modèle réduit est généralement associé à la projection du problème semi-discrétisé sur une base de fonctions spatiales, dont la dimension est réduite par rapport à la dimension de l'espace d'approximation $\mathscr{U}_h^S$. Le point clé réside dans le choix de cette base réduite. La méthode la plus populaire est la méthode de réduction modale, dans laquelle la base réduite est contruite à partir des modes propres de la structure [Qu, 2004]. Plus récemment, des méthodes exploitant une connaissance partielle de la solution du problème de référence ont été proposées [Antoulas, 2005]. La base réduite est alors construite à partir de la POD du vecteur déplacement pris à différentes instants. De nombreuses variantes de ces méthodes ont été développées. Elles sont notamment détaillées dans les ouvrages mentionnés ci-dessus. Dans ce chapitre, on illustre le comportement des méthodes de réduction de modèle les plus simples lorsqu'elles sont appliquées à un problème académique de choc.

3.2 Projection sur une base réduite

En pratique, le problème dont on cherche à réduire le coût de calcul est le problème semi-discrétisé en espace, dont la construction est décrite dans la Section 1.2.2. On rappelle que la solution de ce problème (noté u^h) est cherchée, à chaque instant, dans un espace d'approximation $\mathscr{U}_h^S$ de dimension finie notée n_S.

Les méthodes de réduction de modèle par projection consistent à trouver une approximation de u^h, à chaque instant, dans un sous-espace de $\mathscr{U}_h^S$ dont la dimension M est réduite par rapport à n_S. Ce sous-espace est construit à partir de M modes linéairement indépendants, notés $w_1^h, w_2^h, \cdots$. Ces modes sont calculés dans une étape préliminaire que l'on détaillera dans la suite de cette section. L'espace réduit (noté $\mathscr{U}_h^{\text{rom}}$) est alors défini par

$$\mathscr{U}_h^{\text{rom}} = \left\{ u \in \mathscr{U}_h^S \mid u(x) = \sum_{m=1}^{M} \lambda_m w_m^h(x) \right\}. \tag{3.1}$$

L'approximation de u^h dans $\mathscr{U}_h^{\text{rom}}$ peut s'écrire à chaque instant sous la forme :

$$u^h(x,t) \simeq u_M^h(x,t) = \sum_{m=1}^{M} (\phi(x).\boldsymbol{W}_m)\lambda_m(t)\,, \tag{3.2}$$

où l'on a noté ϕ une base de $\mathscr{U}_h^S$ et $\boldsymbol{W}_m$ les coordonnées du mode w_m^h dans cette base. Les fonctions $\lambda_m(t)$ sont appelées les coordonnées modales. Elles ont ici stockées dans le vecteur $\boldsymbol{\lambda} = [\lambda_1, \cdots, \lambda_M]$. Le problème réduit est alors obtenu en remplaçant l'espace $\mathscr{U}_h^S$ par l'espace réduit $\mathscr{U}_h^{\text{rom}}$ dans la formulation faible du problème.

Problème 3.1. Le problème réduit sur la base $\mathbf{W} = [\boldsymbol{W}_1, \cdots, \boldsymbol{W}_M]$ consiste à trouver le vecteur des coordonnées modales $\boldsymbol{\lambda} : I \to \mathbb{R}^M$ qui vérifient

$$\mathbf{M}^{\text{rom}}.\ddot{\boldsymbol{\lambda}}(t) + \mathbf{K}^{\text{rom}}.\boldsymbol{\lambda}(t) = \boldsymbol{F}^{\text{rom}}(t)\,, \tag{3.3a}$$

$$\text{avec} \quad \boldsymbol{\lambda}(0) = \boldsymbol{U}_0^{\text{rom}}, \tag{3.3b}$$

$$\text{et} \quad \dot{\boldsymbol{\lambda}}(0) = \boldsymbol{V}_0^{\text{rom}}. \tag{3.3c}$$

Les matrices de masse et de raideur du modèle réduit sont respectivement données par $\mathbf{M}^{\text{rom}} = \mathbf{W}'.\mathbf{M}.\mathbf{W}$ et $\mathbf{K}^{\text{rom}} = \mathbf{W}'.\mathbf{K}.\mathbf{W}$. Les projections du vecteur des efforts extérieurs et des vecteurs déplacement initial et vitesse initiale sur la base réduite sont respectivement données par $\boldsymbol{F}^{\text{rom}} = \mathbf{W}'.\boldsymbol{F}$, $\boldsymbol{U}_0^{\text{rom}} = \mathbf{W}'.\boldsymbol{U}_0$ et $\boldsymbol{V}_0^{\text{rom}} = \mathbf{W}'.\boldsymbol{V}_0$.

Le problème réduit est finalement résolu à l'aide d'une méthode d'approximation en temps. En utilisant par exemple une méthode éléments finis en temps, chaque coordonnée $\lambda_m(t)$ peut être approchée sous la forme $\psi(t).\boldsymbol{\Lambda}_m$ où ψ est une base de l'espace d'approximation $\mathscr{U}_{\Delta t}^T$. On aboutit finalement à une approximation de la solution exacte du problème (notée u), sous la forme

$$u(x,t) \simeq u_M^{h,\Delta t} = \phi(x) \otimes \psi(t) \,\underline{\cdot}\, \mathbf{U}_M \quad \text{avec} \quad \mathbf{U}_M = \sum_{m=1}^{M} \boldsymbol{W}_m \otimes \boldsymbol{\Lambda}_m\,. \tag{3.4}$$

L'intérêt d'un tel modèle réduit est de pouvoir diminuer la mémoire nécessaire au stockage de la solution, ainsi que le coût de calcul associé à la résolution de l'équation de mouvement. Bien sûr, l'efficacité de la méthode dépend du choix de la base réduite. Dans la suite de cette section, on compare deux méthodes, classiquement utilisées en dynamique des structures pour construire une base réduite, à savoir la méthode de réduction modale et la méthode dite des instantanés (« snapshot method »).

3.2.1 Méthode de réduction modale

Pour les problèmes de dynamique, la base modale est généralement privilégiée pour construire le modèle réduit, de part la signification physique des modes qui

peut aider l'analyste à construire une base réduite adaptée au problème qu'il souhaite traiter. On considère ici la variante la plus simple de la méthode[1]. On calcule les premiers modes propres de la structure discrétisée (qui sont les vecteurs propres $\boldsymbol{W}_1^{\text{bm}}, \cdots, \boldsymbol{W}_{n_S}^{\text{bm}}$ de la matrice $\mathbf{M}^{-1}.\mathbf{K}$ ordonnés de la valeur propre la plus faible à la plus grande). Puis, on choisit les M modes de la base réduite comme les M premiers modes propres de la structure, c'est-à-dire, $\boldsymbol{W}_m = \boldsymbol{W}_m^{\text{bm}}$ pour $m = 1, \cdots, M$. Dans le cas de la base modale, le problème réduit (3.3a) est un système de M équations découplées qui peut donc être résolu très rapidement.

Lien avec la décomposition en valeurs singulières

Le lien entre les modes propres de la structure et les modes spatiaux associés à la meilleure approximation[2] de rang M a notamment été étudié par [Feeny et Kappagantu, 1998] et [Kerschen et Golinval, 2002]. Dans ces travaux, les auteurs montrent que ces modes sont identiques dans le cas d'un problème de vibration libre (qui correspond à $\boldsymbol{F}(t) = \mathbf{0}$ dans l'équation de mouvement). Par contre, dès lors que le chargement extérieur est non nul, les modes propres de la structure ne coïncident plus avec les modes spatiaux de la meilleure approximation de rang M. Pour illuster cet aspect, on applique la méthode de réduction modale à un problème de vibration libre présenté dans l'Exemple 3.1 ainsi qu'au problème de choc décrit dans l'Exemple 1.1.

Comparaison pour un problème de vibration libre

On considère tout d'abord un problème de vibration libre. La solution exacte de ce problème est présentée dans l'Exemple 3.1. On compare les approximations de rang M de cette solution, obtenues par la méthode de réduction modale et par la meilleure approximation de rang M introduite au chapitre précédent[3]. La comparaison est effectuée avec l'indicateur d'erreur relative totale $\text{err}^{\text{tot}}(M, h, \Delta t)$ définie à l'équation (2.21a).

L'erreur obtenue est présentée sur la Figure 3.1. On observe effectivement que l'approximation de rang M construite avec la base modale, et la meilleure approximation de rang M construite avec la SVD de la solution discrète, convergent toutes les deux vers la même décomposition. Les modes en espace et en temps obtenus par les deux méthodes sont présentés sur la Figure 3.2. On observe que les modes propres de la structure ainsi que leurs coordonnées modales coïncident respectivement avec les modes en espace et en temps associés à la meilleure approximation de rang M. Sur

1. Une alternative est par exemple la méthode des vecteurs de Ritz qui permet d'améliorer la précision de la base modale en introduisant des modes dépendant du chargement[Qu, 2004].

2. On rappelle que la meilleure approximation de $u^{h,\Delta t}$ dans $\mathscr{R}_{M,h,\Delta t}$ est construite avec la SVD de $\mathbf{U}$ tronquée au rang M.

3. L'espace d'approximation $\mathscr{U}_h^S$ est construit par la méthodes des éléments finis P2 et le problème est résolu en temps avec le schéma TDG P2.

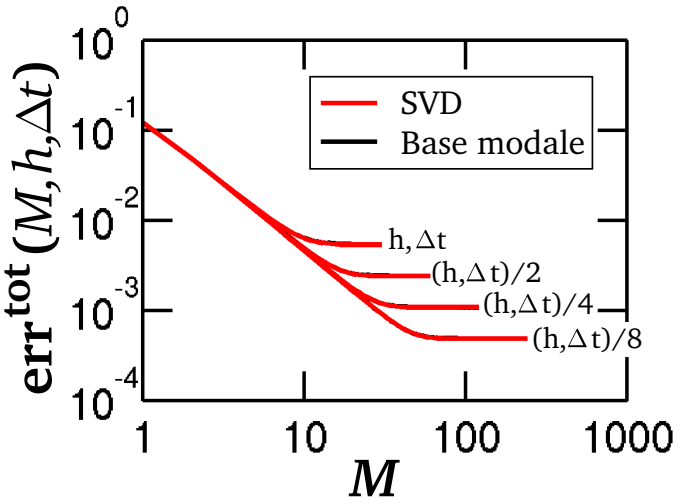

FIGURE 3.1: Erreur entre la solution exacte et la solution du modèle réduit en fonction du nombre M de modes, pour le problème de vibration libre de l'Exemple 3.1.

la Figure 3.2, on observe une différence entre les deux méthodes, seulement pour les modes $m \geq 16$ à partir desquels l'erreur de décomposition devient inférieure à l'erreur de discrétisation. On observe pour les deux méthodes, que les modes en espace et en temps convergent vers la solution analytique décrite dans l'Exemple 3.1 lorsque l'on augmente la dimension de l'espace d'approximation.

Comparaison pour un problème de choc

On effectue la même comparaison dans le cas d'un problème de choc. On reprend le cas test de l'Exemple 1.1 avec différentes valeurs de κ et pour $L = 1\mathrm{m}$, $T = 1\mathrm{s}$ et $c = 1\mathrm{m/s}$. L'erreur obtenue pour $\kappa = 0.1 - 1 - 10 - 100$ est présentée sur la Figure 3.3. Pour toutes les valeurs de κ, le nombre de modes nécessaires pour obtenir une précision de 0.1% est environ un ordre de grandeur plus grand avec la méthode de réduction modale qu'avec la meilleure approximation de rang M. Pour $\kappa = 100$, il faut par exemple 1413 modes propres de la structure pour obtenir une solution espace-temps précise à 0.1%, alors qu'il faut seulement 127 modes espace-temps de la meilleure approximation de rang M pour obtenir la même précision.

Intuitivement, on aurait pu penser que l'erreur obtenue pour un problème basse fréquence est similaire pour les deux méthodes. Or, on observe pour les valeurs de $\kappa = 0.1 - 1$, que la meilleure approximation de rang M est beaucoup plus précise que la méthode de réduction modale. On observe à l'opposé, que plus la valeur de κ augmente et plus l'erreur obtenue pour les premiers modes est similaire. Puis, à partir d'un certain nombre de modes, l'erreur entre les deux méthodes diffèrent complètement (voir les courbes sur la Figure 3.3 pour $\kappa = 100$ à partir de $M \simeq 30$). On retrouve ce comportement en comparant les modes obtenus par les deux méthodes (voir la Figure

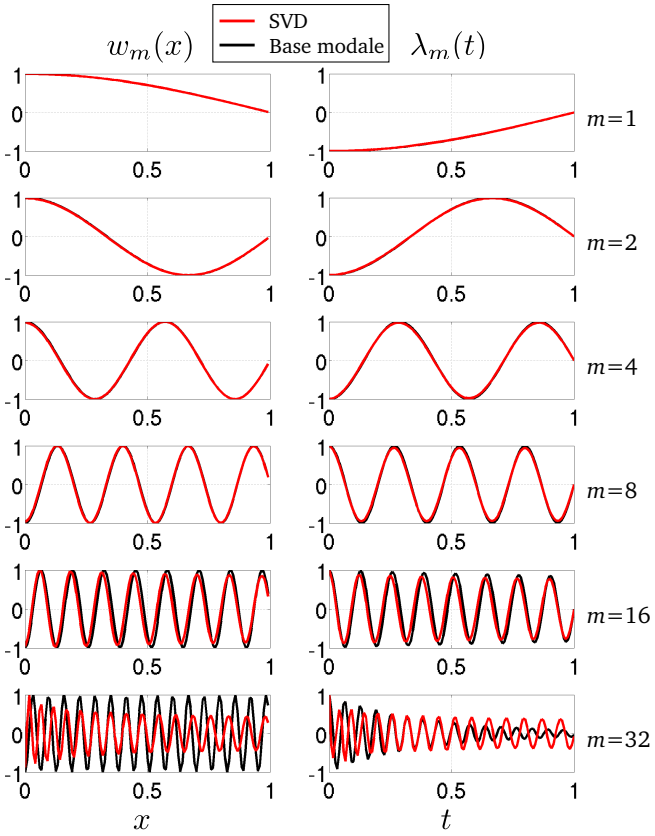

FIGURE 3.2: Modes en espace et en temps obtenus dans le cas du problème de vibration libre décrit dans l'Exemple 3.1.

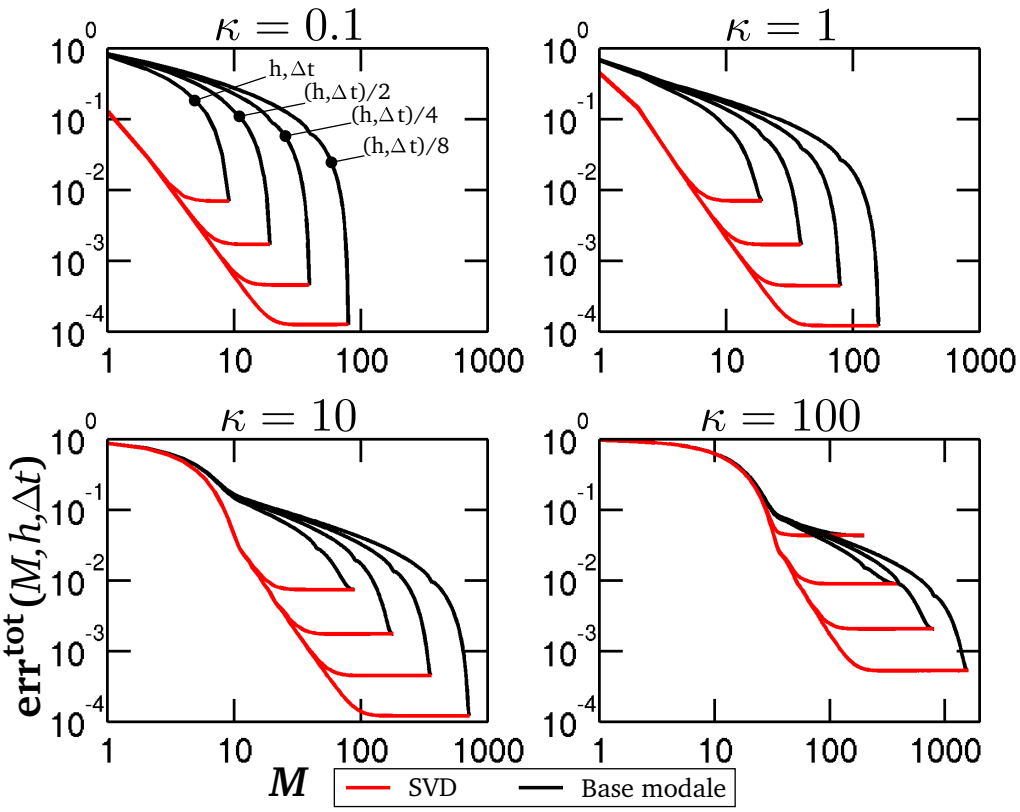

FIGURE 3.3: Erreur entre la solution exacte et la solution du modèle réduit en fonction du nombre M de modes, pour le problème de choc de l'Exemple 1.1.

3.4). Pour $\kappa = 100$, les premiers modes sont assez similaires (voir les modes $m = 1$ à 16), alors que les modes associées aux plus hautes fréquences sont très différents (voir les modes $m = 32$ à 128).

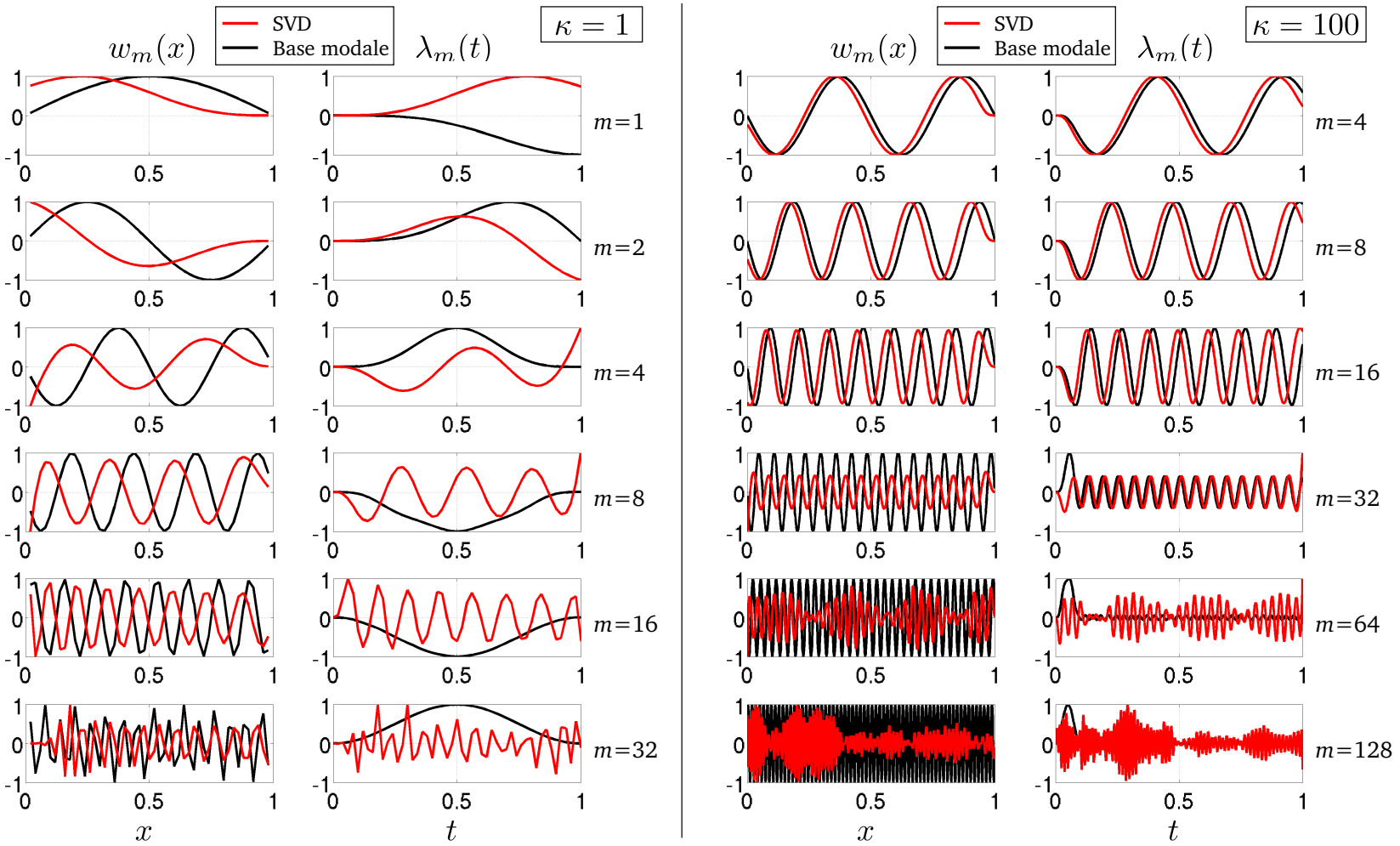

FIGURE 3.4: Modes en espace et en temps obtenus dans le cas du problème de choc décrit dans l'Exemple 1.1 avec $\kappa = 1$ et 100.

Exemple 3.1. (Solution analytique d'un problème de vibration libre) Dans cet exemple, on montre comment construite la solution analytique d'un problème de vibration libre par la méthode de séparation de variables. On considère ici une poutre de longueur L encastrée au point $x = L$ et libre au point $x = 0$. Cette poutre est soumise à un déplacement initial de type compression. On s'intéresse à la réponse transitoire de la poutre sur l'intervalle de temps $I = [0, T]$. Dans ce cas, la modélisation du problème consiste à trouver le déplacement $u(x,t)$ en tout point $(x,t) \in [0,L] \times [0,T]$ qui vérifie :

$$\frac{\partial^2 u}{\partial x^2} = \frac{1}{c^2}\frac{\partial^2 u}{\partial t^2} \quad \text{avec} \quad \begin{cases} \frac{\partial u}{\partial x}(0,t) & = 0 \\ u(L,t) & = 0 \\ u(x,0) & = u_0(x) = U_0(x/L-1) \\ \partial u/\partial t(x,0) & = 0 \end{cases}. \tag{3.5}$$

La solution exacte de ce problème (obtenue par la méthode des caractéristiques) est représentée sur la Figure 3.5. La méthode de séparation de variables permet de trouver analytiquement une représentation à variables séparées espace-temps de cette solution [Zauderer, 1989]. On suppose tout d'abord que la solution peut s'écrire sous la forme $u(x,t) = w(x)\lambda(t)$. Puis on remplace cette expression de u dans l'équation d'équilibre de façon à la découpler. En divisant par $w\lambda$, on obtient les deux équations suivantes,

$$\frac{\mathrm{d}^2 w(x)}{\mathrm{d}x^2} + \frac{\omega^2}{c^2} w(x) = 0\,, \tag{3.6a}$$

$$\frac{\mathrm{d}^2 \lambda(t)}{\mathrm{d}t^2} + \omega^2 \lambda(t) = 0\,, \tag{3.6b}$$

qui sont simplement reliées par l'intermédiaire de la constante ω. Avec les conditions aux limites introduites précédemment, on montre que la solution de l'équation (3.6a) est donnée sous forme générale par,

$$w_n(x) = \cos(\frac{\omega_n}{c}x) \quad \text{avec} \quad \omega_n = \frac{(2n-1)\pi}{2}\frac{c}{L} \quad \text{pour } n \geq 1\,. \tag{3.7}$$

En remplaçant ω par ω_n dans l'équation (3.6b), on montre que la solution générale de l'équation (3.6b) s'écrit sous la forme suivante,

$$\lambda_n(t) = \alpha_n \cos(\omega_n t) + \beta_n \sin(\omega_n t) \quad \text{pour } n \geq 1\,. \tag{3.8}$$

La solution générale du problème de vibration libre est alors exprimée sous la forme,

$$u(x,t) = \sum_{m=1}^{\infty} w_m(x)\lambda_m(t)\,. \tag{3.9}$$

Il reste à déterminer les coefficients α_n et β_n tels que cette solution vérifie les conditions initiales. Pour cela, on orthonormalise les fonctions w_n au sens d'un certain

produit scalaire, que l'on note $\langle , \rangle$. Puis, on obtient les coefficients en projetant les conditions initiales sur la base des fonctions $w_1, w_2, \cdots$. On obtient dans le cas traité ici, $\alpha_n = \langle u_0 , w_m \rangle$ et $\beta_n = 0$ pour tout $n \geq 1$.

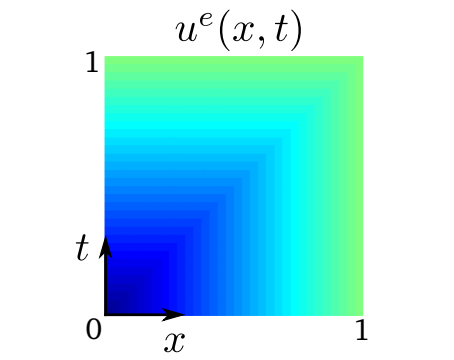

FIGURE 3.5: Solution exacte du problème de vibration libre avec $L = 1\text{m}$, $T = 1\text{s}$ et $c = 1\text{m/s}$.

3.2.2 Méthode POD-Snapshot

Plus récemment, des méthodes de réduction de modèle basées sur une connaissance partielle de la solution ont été proposées. Le principe est de construire la base réduite à partir d'un nombre fini d'instantannés (appelés en anglais « snapshots ») qui sont obtenus soit à partir de mesures expérimentales, soit en résolvant le problème semi-discrétisé à différents instants. On utilise alors la décomposition orthogonale propre (POD) de ces instantannés pour construire la base réduite. Les modes spatiaux sont sélectionnés de façon à ce que le modèle réduit est un contenu énergétique équivalent à celui du modèle complet. Ces méthodes de réduction de modèle sont appelées « a posteriori » car la base réduite est construite dans une première étape, précédent le calcul du modèle réduit. Tout d'abord introduite en mécanique des fluides [4], les méthodes de réduction de modèle a posteriori ont été plus récemment appliquées en dynamique des structures [5]. Une nouvelle fois, de nombreuses variantes existes [Antoulas, 2005] et on ne considère ici qu'une approche simplifiée.

Méthode des snapshots

La méthode des snapshots consiste à résoudre le problème de référence (appelé « Full Order Model (FOM) ») en un nombre réduit d'instants $0 \leq t'_1 < t'_2 < \cdots < t'_{M_s} \leq T$. Le vecteur déplacement calculé à ces différents instants est stocké dans la « matrice des snapshots », notée $\mathbf{S} = [\boldsymbol{U}(t'_1), \cdots, \boldsymbol{U}(t'_{M_s})] \in \mathbb{R}^{n_S} \otimes \mathbb{R}^{M_s}$. On construit alors la base réduite avec les M premiers vecteurs singuliers à gauche de la matrice des snapshots. En notant, $\boldsymbol{W}_1^{\text{snap}}, \cdots, \boldsymbol{W}_{\text{rang}(\mathbf{S})}^{\text{snap}}$ les vecteurs singuliers à gauche de $\mathbf{S}$, on choisit $\boldsymbol{W}_m = \boldsymbol{W}_m^{\text{snap}}$ pour $m = 1, \cdots, M$ avec $M \leq \text{rang}(\mathbf{S})$. La résolution temporelle du model réduit donne finalement les modes en temps $\boldsymbol{\Lambda}_m$, pour $m = 1, \cdots, M$.

Remarque 3.1. *La meilleure approximation de rang M correspond au cas limite où la matrice des snapshots contient le vecteur déplacement pris à tous les instants de la partition de référence du domaine temporel, c'est-à-dire $\mathbf{S} = \mathbf{U} \in \mathbb{R}^{n_S} \otimes \mathbb{R}^{n_T}$ où n_T est la dimension de l'espace d'approximation en temps.*

4. On pourra consulter les travaux précurseurs de [Berkooz *et al.*, 1993]. On pourra également consulter la revue récente de [Lassila *et al.*, 2013] où les auteurs donnent de nombreux conseils et explications concernant l'efficacité de la méthode. Notamment, on pourra consulter dans cette revue, la section intitulée « Never try to reduce the irreducible » qui donne des arguments pour montrer qu'une approche a posteriori n'est pas adaptée pour réduire des problèmes de propagation d'ondes. Pour ce type de problèmes, ils expliquent que l'utilisation d'un maillage grossier en temps pour calculer les snapshots, implique que les valeurs propres de la matrice de corrélation décroissent très lentement. Cet argument justifie cependant l'utilisation d'une méthode de réduction de modèle a priori où le maillage utilisé pour calculer le modèle réduit est le même que celui utilisé pour calculer la solution de référence.

5. On pourra consulter la revue de [Kerschen *et al.*, 2005] ou les applications plus récentes proposées par [Placzek *et al.*, 2008, Bamer et Bucher, 2012, Eftekhar Azam et Mariani, 2013], ou encore le commentaire de [Glösmann et Kreuzer, 2009] sur l'efficacité de la méthode.

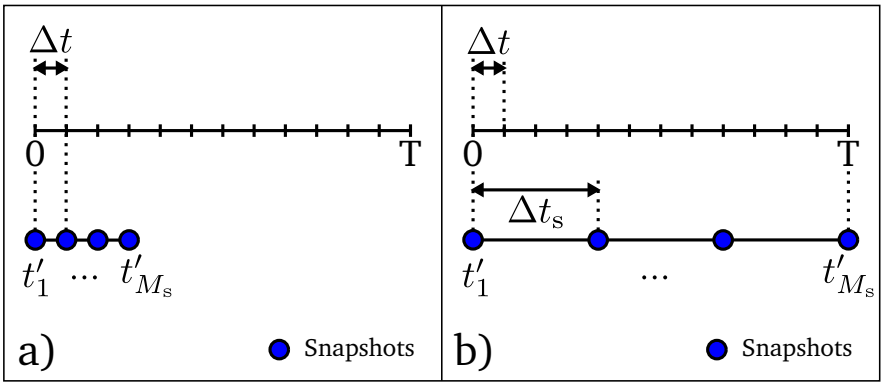

FIGURE 3.6: Méthodes des snapshots : a) construction sur les premiers instants, et b) construction sur un maillage grossier en temps.

Stratégies pour le calcul des snapshots

La précision du modèle réduit dépend du nombre de snapshots (noté M_s), du nombre de modes dans la base réduite (noté M), ainsi que de la stratégie qui est utilisée pour calculer les snapshots. On compare ici deux méthodes, schématisées sur la Figure 3.6. Dans la première méthode (notée méthode $t_1\,t_2$), les snapshots sont calculés aux M_s premiers instants du maillage temporel de référence. Dans la deuxième méthode (notée méthode Δt), ils sont calculés à M_s instants pris sur un maillage grossier en temps. Dans ce cas, on note Δt_s le pas de temps de ce maillage grossier.

Pour comparer ces deux méthodes, on reprend le problème de choc de l'Exemple 1.1 avec différentes valeurs de κ et pour $L = 1$m, $T = 1$s et $c = 1$m/s. On rappelle que le problème est discrétisé en espace avec des éléments finis P2 et résolu en temps avec le schéma TDG P2. On fixe ici les paramètres du maillage de façon à ce que l'erreur relative de discrétisation (notée err^{disc}, voir l'équation (2.21b)) soit égale à 0.1% pour tous les cas tests. L'erreur due à une approximation de rang M est calculée par rapport à la solution exacte du problème de référence (on calcule l'erreur relative entre u et $u_M^{h,\Delta t}$, notée err^{tot}, voir l'équation (2.21c)). On compare alors, pour un rang M donné, la valeur de err^{tot} obtenue avec les méthodes des snapshots à celle obtenue avec la meilleure approximation de rang M.

Méthode $t_1\,t_2$

La matrice des snapshots calculés aux premiers instants du maillage temporel de référence, est représentée sur la Figure 3.7 pour le cas test à $\kappa = 10$ et pour différents choix du temps t'_{M_s}. Avec cette stratégie, les snapshots sont calculés avec la même précision que le modèle de référence. Cependant, on ne voit que les premiers évènements

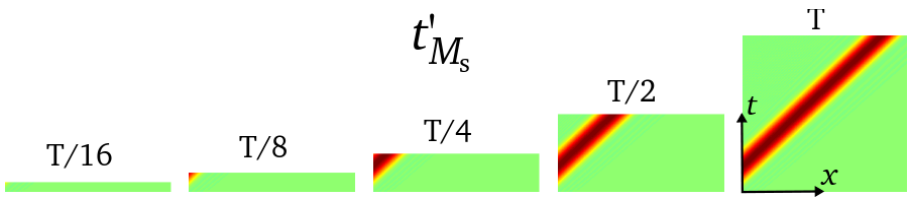

FIGURE 3.7: Matrice des snapshots calculés sur les premiers pas de temps ($\kappa = 10$).

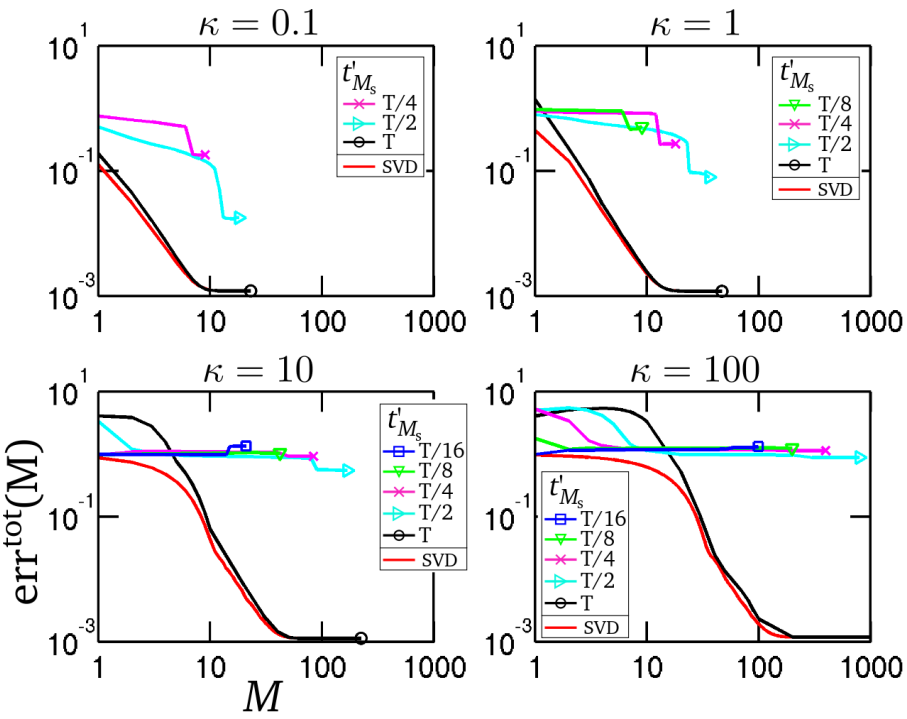

FIGURE 3.8: Erreur entre la solution exacte et une approximation de rang M dans le cas où les snapshots sont calculés sur les premiers pas de temps

se produisant au cours de la simulation [6]. La valeur de l'erreur due à une approximation de rang M (err^{tot}) est représentée en fonction du rang M sur la Figure 3.8, pour les différents cas tests et différents choix du temps t'_{M_s}. On observe que le modèle réduit échoue à représenter précisémment la solution de référence dans tous les cas où $t'_{M_s} \neq T$.

On peut expliquer cet échec en regardant la définition de κ (on rappelle que $\kappa = (\frac{L}{c\Delta T})^2$). Comme la durée de la simulation est choisie égale au temps que l'onde met pour parcourir une fois la longueur L, choisir une durée $t'_{M_s} < T$ revient à diminuer la longueur L (un point du domaine spatial situé après la distance $c\,t'_{M_s}$ n'est pas sollicité). Les snapshots sont donc représentatifs du champ de déplacement associé à une valeur de κ plus faible que celle du problème de référence.

Méthode Δt

La matrice des snapshots calculés sur un maillage grossier en temps, est représentée sur la Figure 3.9 pour le cas test à $\kappa = 10$ et pour différents choix du pas de temps Δt_s. Avec cette stratégie, la matrice des snapshots est moins précise que le modèle de référence. On peut cependant voir tous les évènements (d'une durée caractéristique supérieure à Δt_s) se produisant au cours de la simulation. La valeur de l'erreur (err^{tot}) est représentée en fonction du rang M sur la Figure 3.10, pour les différents cas tests et différents choix du pas de temps Δt_s. Les résultats obtenus avec cette stratégie sont bien meilleurs que ceux obtenus avec la stratégie $t_1\,t_2$. On observe que le modèle réduit est assez proche de la meilleure approximation de rang M et ce pour des valeurs relativement grossières de Δt_s. Ces résultats peuvent être expliqués par le fait que lorsque l'on calcule la POD de la solution d'un problème donné sur deux maillages différents, on observe que les premiers modes sont très similaires entre les deux maillages.

Pour un choix de Δt_s donné, on observe que le modèle réduit de rang maximal est plus précis que la solution donnée par les snapshots (calculée sur le maillage grossier en temps). Par exemple, pour le cas test à $\kappa = 100$ et le choix $\Delta t_s = 16\Delta t$, l'erreur de discrétisation pour la solution grossière est $\text{err}^{\text{disc}}(h, \Delta t_s) = 31\%$ et l'erreur du modèle réduit de rang maximal est $\text{err}^{\text{tot}}(M_s, h, \Delta t) = 2.9\%$. La projection sur la base réduite, puis la résolution sur un maillage fin en temps, améliore donc la précision d'une solution calculée sur un maillage grossier en temps [7].

Cependant, on doit calculer les snapshots sur une maillage en temps quasiment aussi fin que le maillage de référence si l'on souhaite obtenir un modèle réduit aussi précis que la solution discrète de référence. Il faut en effet choisir $\Delta t_s = 2\Delta t$ si on veut avoir $\text{err}^{\text{tot}}(M_s, h, \Delta t) \simeq \text{err}^{\text{disc}}(h, \Delta t) (= 0.1\%)$.

6. Typiquement, on ne verra pas un chargement extérieur survenant après le temps $t'_{M_s} = (M_s - 1)\Delta t$ où Δt est le pas de temps du modèle de référence.

7. On pourrait également appliquer cette stratégie dans le cas d'une solution grossière en espace et en temps.

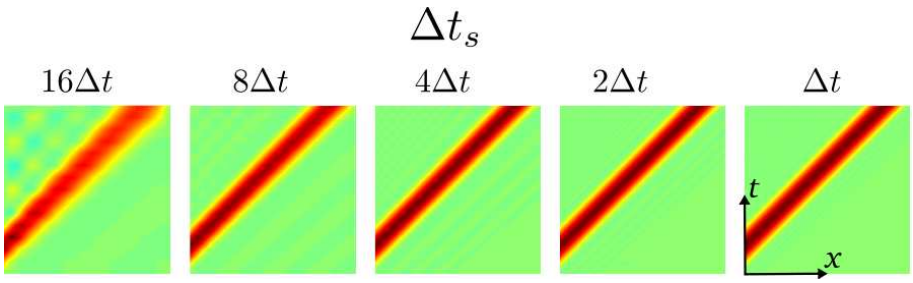

FIGURE 3.9: Matrice des snapshots calculés sur un maillage grossier en temps ($\kappa = 10$).

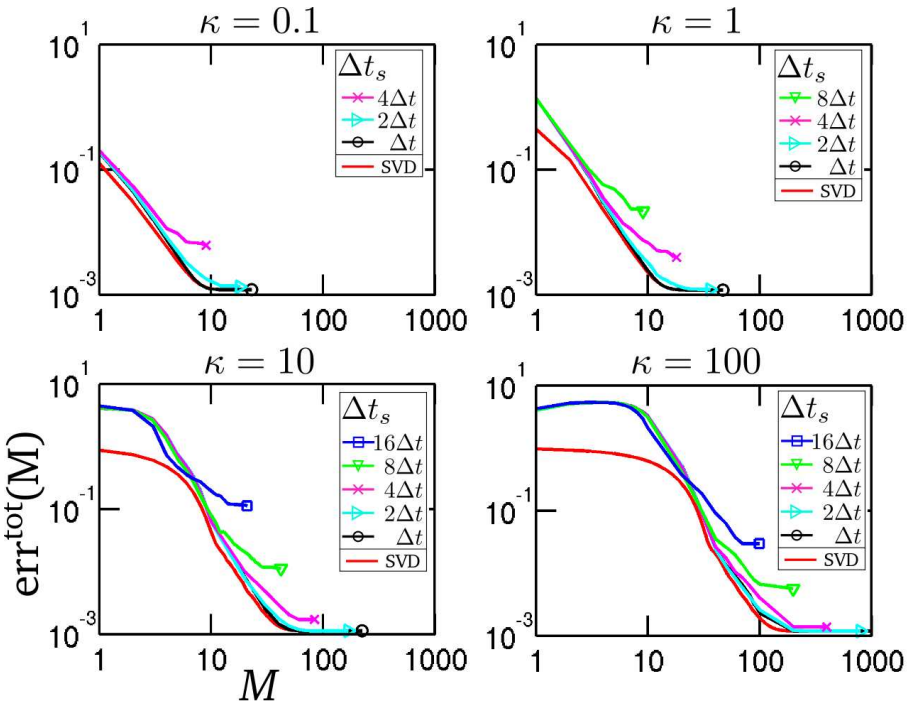

FIGURE 3.10: Erreur entre la solution exacte et une approximation de rang M dans le cas où les snapshots sont calculés sur un maillage grossier en temps

3.3 Conclusion

Dans ce chapitre, les méthodes de réduction de modèle classiquement utilisées en dynamique des structures ont été présentées. Ces méthodes reposent sur la projection du problème semi-discrétisé sur une base de fonctions spatiales dont la dimension est réduite par rapport à la dimension de l'espace d'approximation. Le modèle réduit est ensuite résolu en temps. On aboutit ainsi à une approximation de la solution de référence sous la forme d'une représentation à variables séparées espace-temps.

La précision de ces méthodes dépend du choix de la base réduite. On a comparé dans ce chapitre les bases réduites les plus populaires et les plus simples, à savoir la base modale, et la base extraite de la POD du vecteur déplacement pris à différents instants (dite méthode des snapshots). Les résultats suivants ont été obtenus :

- Pour un problème de vibration libre, la base modale est optimale (au sens de la meilleure approximation de rang M). Pour un problème de choc, ce n'est cependant plus le cas et la meilleure approximation de rang M est beaucoup plus précise (de un à trois ordres de grandeur pour les cas testés et en fonction du rang M).
- Pour un problème de choc, la base réduite construite à partir de la POD du vecteur déplacement pris à différents instants, est représentative de la solution du problème de référence seulement si les instantanés sont calculés sur un maillage grossier du domaine temporel. Dans ce cas, la méthode permet de construire un modèle réduit peu précis par rapport à la précision de la solution discrète de référence.

Pour les problèmes de choc, il s'avère donc que **les méthodes classiques de réduction de modèle échouent à construire une bonne approximation de la meilleure approximation de rang** M, sans avoir à calculer ni stocker, soit la quasi totalité des modes structuraux (pour la méthode de réduction modale), soit une solution sur un maillage temporel presqu'aussi fin que le maillage temporel de référence (pour la méthode POD-Snapshots).

Aussi, de nouvelles stratégies doivent être proposées. Dans la suite du manuscrit, on se tourne vers les méthodes de réduction de modèle dites a priori, qui ne nécessitent aucune connaissance supplémentaire sur la solution de référence que les opérateurs du problème espace-temps dont elle est solution. Dans l'esprit des méthodes de réduction de modèle par projection sur une base de fonctions spatiales, une stratégie consiste à construire de façon adaptative, une base réduite enrichie à l'aide d'un sous-espace de Krylov [Ryckelynck, 2005, Ryckelynck *et al.*, 2006]. On se concentre dans ce manuscrit sur un autre type de méthodes de réduction de modèle a priori, où les modes en espace et en temps sont les inconnues du problème [Ladevèze, 1999, Chinesta *et al.*, 2011]. Ce type de méthodes exploite la structure tensorielle des opérateurs du problème espace-temps. La construction de ces opérateurs, dans un format adapté, est l'objet du prochain chapitre.

Chapitre 4

Représentation du problème d'élastodynamique sous format tensoriel

Dans ce chapitre, on montre comment écrire le problème d'élastodynamique, discrétisé en espace et en temps, sous la forme d'un unique système linéaire dont les opérateurs sont donnés sous format tensoriel.

Sommaire

4.1 Principe

La discrétisation en espace et en temps du problème d'élastodynamique aboutit classiquement à un schéma incrémental, qui permet de calculer les coordonnées du champ de déplacement dans une base d'approximation spatiale, à chaque instant. Ces coordonnées sont stockées dans un tenseur du second ordre, que l'on note ici $\mathbf{u} \in \mathbb{R}^{n_S} \otimes \mathbb{R}^{n_T}$. Aussi, le processus incrémental consiste à calculer chaque colonne de $\mathbf{u}$, l'une après l'autre en résolvant n_T fois, un système linéaire de taille $n_S \times n_S$. Dans ce chapitre, on montre **comment écrire le problème discrétisé en espace et en temps sous la forme d'un unique système linéaire**, que l'on appellera le « problème espace-temps » et dont la résolution donne le tenseur $\mathbf{u}$ en une seule fois. Bien sûr, une résolution brutale du problème espace-temps (à l'aide d'un solveur classique de système linéaire) n'est pas envisageable puisqu'elle nécessite de résoudre un sytème linéaire de taille $n \times n$ avec $n = n_S n_T$, ce qui est beaucoup plus coûteux qu'une résolution incrémentale.

Cependant, on peut exploiter la structure tensorielle du problème espace-temps pour réduire la complexité des opérations algébriques usuelles. C'est la construction de cette représentation tensorielle du problème espace-temps que l'on illustre dans ce chapitre. On verra alors dans le prochain chapitre, comment tirer parti de ce format pour implémenter des solveurs génériques, donnant une approximation de la solution du problème espace-temps, à moindre coût.

4.1.1 Problème à un champ

L'objectif de ce chapitre est d'aboutir à une représentation des opérateurs du problème espace-temps sous la forme de l'équation (4.1). Le formalisme utilisé dans ce chapitre est décrit en détail dans l'Annexe A.

$$\mathbf{A} \overset{\mathrm{D}}{\cdot} \mathbf{u} = \mathbf{b}, \tag{4.1}$$

où les opérateurs $\mathbf{A} \in \mathbb{R}^{n_S} \otimes \mathbb{R}^{n_S} \otimes \mathbb{R}^{n_T} \otimes \mathbb{R}^{n_T}$ et $\mathbf{b} \in \mathbb{R}^{n_S} \otimes \mathbb{R}^{n_T}$ sont donnés sous format tensoriel par

$$\mathbf{A} = \sum_{k=1}^{M_A} \mathbf{A}_k^S \otimes \mathbf{A}_k^T \quad \text{et} \quad \mathbf{b} = \sum_{k=1}^{M_b} \boldsymbol{b}_k^S \otimes \boldsymbol{b}_k^T. \tag{4.2}$$

Ce type de représentation des opérateurs est classiquement utilisé pour formuler des problèmes d'équations aux dérivées partielles sur des espaces produits tensoriels. Il a notamment été introduit par [Friedman et Kline, 1955] dans le contexte de la méthode

de séparation de variables. Il est également utilisé par [Cannarozzi et Mancuso, 1995] pour formuler le problème d'élastodynamique sur le domaine espace-temps.

Limitations

Les principales limitations associées à l'écriture du problème espace-temps sous la forme de (4.2) sont les suivantes :

- le maillage espace-temps doit être structuré [1] [2],
- le domaine spatial et la partition de sa frontière ne doivent pas évoluer au cours du temps [3],
- la formulation faible du problème sur le domaine espace-temps doit pouvoir s'écrire sous la forme d'une somme de produits d'intégrales respectivement définies sur l'espace et sur le temps [4] [5].

4.1.2 Problème multichamps

Dans le cas d'un problème à F champs [6] [7] [8], on peut généraliser l'équation (4.1) sous la forme de l'équation (4.3) en introduisant la notion de $F-$tuple de tenseurs d'ordre D (définie dans l'Annexe A).

1. Ceci n'est pas très contraignant puisque le domaine espace-temps est généralement construit par « extrusion » du domaine spatial le long du domaine temporel. Cependant, l'utilisation d'un espace produit tensoriel implique que le maillage ne peut pas être rafiné localement dans le domaine espace-temps, comme c'est le cas dans les stratégies espace-temps développées dans [Aubry *et al.*, 1999, Cavin *et al.*, 2005, Abedi *et al.*, 2006].

2. On peut cependant coupler une approximation à variables séparées avec méthodes des classiques d'approximation pour rafiner localement le maillage [Ammar *et al.*, 2011] ou introduire une discontinuité [Niroomandi *et al.*, 2012, Giner *et al.*, 2013].

3. On pourra cependant consulter [Leygue et Verron, 2010, Nouy *et al.*, 2011, Ammar *et al.*, 2013a, Ammar *et al.*, 2013b] qui proposent différentes stratégies permettant de traiter le cas où la partition des frontières et la géométrie du domaine de calcul dépendent de paramètres.

4. On notant $a(,)$ une forme bilinéaire sur $\mathscr{U}^S \otimes \mathscr{U}^T$, on doit pouvoir écrire $a(w^\star \lambda^\star, w\lambda) = \sum_{k=1}^{M_A} a_k^S(w^\star, w) a_k^T(\lambda^\star, \lambda)$ où $a_k^S(,)$ et $a_k^T(,)$ sont des formes bilinéaires respectivement sur $\mathscr{U}^S$ et $\mathscr{U}^T$. Et de même pour la forme linéaire.

5. Si une telle décomposition n'est pas connue a priori, on peut la calculer a posteriori (voir par exemple [Van Loan et Pitsianis, 1992, Beylkin et Mohlenkamp, 2002, Hackbusch, 2012]). Cependant, les rangs M_A ou M_B doivent être suffisamment faibles pour que la résolution du problème sous format tensoriel soit efficace.

6. Ces champs peuvent être associés à différentes physiques (par exemple le déplacement et la vitesse) ou bien, ils peuvent être les composantes d'un champ de vecteurs.

7. Tous les champs sont définis sur le domaine espace-temps $\Omega \times I$.

8. Chaque champ peut être pris dans un espace d'approximation de dimension différente, c'est-à-dire $\mathbf{u}_i \in \mathbb{R}^{n_S(i)} \otimes \mathbb{R}^{n_T(i)}$ pour $i = 1, \cdots, F$ où on peut avoir $n_S(i) \neq n_S(j)$ et $n_T(i) \neq n_T(j)$ pour $i \neq j$.

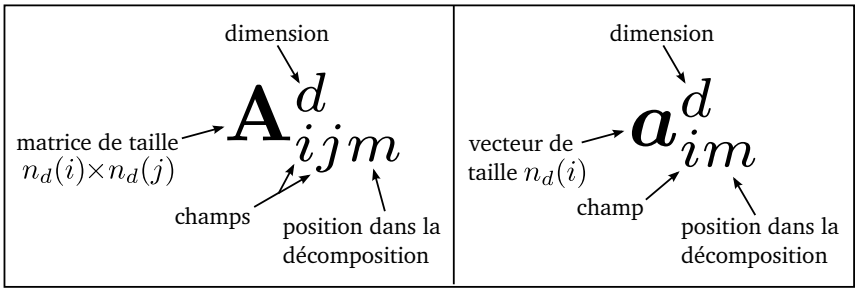

FIGURE 4.1: Définitions des indices des composantes des tuples $[\![\mathbf{A}]\!]$ et $[\mathbf{a}]$ données sous format séparé par $\mathbf{A}_{ij} = \sum_{m=1}^{M_A(i,j)} \mathbf{A}^S_{ijm} \otimes \mathbf{A}^T_{ijm}$ et $\mathbf{a}_i = \sum_{m=1}^{M_a(i)} \boldsymbol{a}^S_{im} \otimes \boldsymbol{a}^T_{im}$.

$$[\![\mathbf{A}]\!] \overset{\mathrm{D}}{\cdot} [\mathbf{u}] = [\mathbf{b}] \quad \Leftrightarrow \quad \begin{cases} \mathbf{A}_{11} \overset{\mathrm{D}}{\cdot} \mathbf{u}_1 + \cdots + \mathbf{A}_{1F} \overset{\mathrm{D}}{\cdot} \mathbf{u}_F &= \mathbf{b}_1 \\ \qquad\qquad\qquad\qquad\qquad\quad | & \\ \mathbf{A}_{F1} \overset{\mathrm{D}}{\cdot} \mathbf{u}_1 + \cdots + \mathbf{A}_{FF} \overset{\mathrm{D}}{\cdot} \mathbf{u}_F &= \mathbf{b}_F \end{cases}, \tag{4.3}$$

où les opérateurs $\mathbf{A}_{ij} \in \mathbb{R}^{n_S(i)} \otimes \mathbb{R}^{n_S(j)} \otimes \mathbb{R}^{n_T(i)} \otimes \mathbb{R}^{n_T(j)}$ et $\mathbf{b}_i \in \mathbb{R}^{n_S(i)} \otimes \mathbb{R}^{n_T(i)}$ sont donnés sous format tensoriel pour $i, j = 1, \cdots, F$ par

$$\mathbf{A}_{ij} = \sum_{k=1}^{M_A(i,j)} \mathbf{A}^S_{ijk} \otimes \mathbf{A}^T_{ijk} \quad \text{et} \quad \mathbf{b}_i = \sum_{k=1}^{M_b(i)} \boldsymbol{b}^S_{ik} \otimes \boldsymbol{b}^T_{ik}. \tag{4.4}$$

Remarque 4.1. *Le système linéaire (4.3) généralise le système linéaire (4.1), autrement dit, le système (4.3) n'est pas un cas particulier du système (4.1).*

4.1.3 Stratégies de résolution

On trouve dans la littérature de nombreuses stratégies pour résoudre les systèmes linéaires (4.1) et (4.3). On peut distinguer deux grandes classes de stratégies.

Dans la première, l'opérateur $\mathbf{b}$ est donné sous format non séparé et le système linéaire (4.1) est écrit sous la forme d'un problème d'équations matricielles linéaires (« linear matrix equations ») [Ding et Chen, 2005] :

$$\mathbf{A} \overset{\mathrm{D}}{\cdot} \mathbf{u} = \mathbf{b} \quad \Leftrightarrow \quad \sum_{k=1}^{M_A} \mathbf{A}^S_k . \mathbf{u} . (\mathbf{A}^T_k)' = \mathbf{b}, \tag{4.5}$$

également appelé équations de Sylvester généralisées (« generalized Sylvester equations ») dans le cas où $M_A = 2$. Le système linéaire (4.3) correspond à un problème d'équations matricielles linéaires couplées (« coupled linear matrix equations »)[Ding et Chen, 2006], écrit dans une forme aussi générale que celle proposée

dans [Zhou *et al.*, 2009] [9]. Une très vaste littérature existe pour résoudre ce type de problème, notamment dans la communauté du contrôle de systèmes. Cependant, ces méthodes donnent la solution du système linéaire sous format non-séparé. Et le stockage de cette solution (de même que le membre de droite) n'est pas possible dès lors que la dimension du problème espace-temps est trop grande.

Aussi, on s'intéresse dans le cadre de ce manuscrit, à une deuxième classe de méthode, où les opérateurs **A** et **b** ainsi que la solution **u** sont donnés sous format séparé. L'objectif est de ne jamais avoir à stocker les tenseurs **A**, **u** ou **b** (ou une autre quantité [10]) sous format non séparé. Dans certains cas particuliés, la solution sous format séparé peut être obtenue de façon directe [Friedman et Kline, 1955, Lynch *et al.*, 1964]. Dans le cas général, des méthodes itératives sont utilisées. On peut alors distinguer une nouvelle fois deux classes de méthodes. Dans la première, l'idée générale est d'utiliser les solveurs itératifs classiques (écrits sous format tensoriel) et de tronquer l'itéré courant à un rang donné [Kressner et Tobler, 2010, Khoromskij et Schwab, 2011, Matthies et Zander, 2012, Ballani et Grasedyck, 2013, Giraldi *et al.*, 2013]. Dans ces stratégies, les itérations sont menées jusqu'à ce que le système linéaire soit résolu à précision machine. Aussi, une autre classe de méthodes consistent à calculer seulement une approximation de la solution sous format séparé [Ladevèze, 1999, Beylkin et Mohlenkamp, 2005, Ammar *et al.*, 2006, Nouy, 2007, Chinesta *et al.*, 2008, Espig *et al.*, 2012]. C'est dans l'esprit de ces stratégies que l'on se place dans la suite du manuscrit.

Remarque 4.2. *Dans le cas d'un problème de quasi-statique, l'opérateur* **A** *du problème espace-temps est de rang un. Aussi, le système linéaire s'écrit sous la forme,*

$$(\mathbf{K}\otimes\mathbf{I})\,\dot{\underline{D}}\mathbf{u} = \sum_{k=1}^{M_b} \boldsymbol{b}_k^S \otimes \boldsymbol{b}_k^T\,, \tag{4.6}$$

où **K** *est la matrice de raideur et* **I** *la matrice identité en temps. Lorsque de plus, le rang* $M_b = 1$*, la solution* **u** *s'écrit exactement sous la forme d'une décomposition de rang un, donnée par*

$$\mathbf{u} = \boldsymbol{w}\otimes\boldsymbol{\lambda} \quad avec \quad \boldsymbol{w} = \mathbf{K}^{-1}.\boldsymbol{b}^S \; et \; \boldsymbol{\lambda} = \boldsymbol{b}^T\,. \tag{4.7}$$

On peut alors imaginer différentes stratégies basées sur ce principe, pour trouver très rapidement la meilleure approximation de rang M *de* **u** *lorsque* $M_b > 1$ *et que* **A** *est de rang un.*

9. La publication de [Zhou *et al.*, 2009] est en accord avec la Remarque 4.1.

10. Notamment, le stockage du résidu dans un algorithme itératif peut poser problème (voir le Chapitre 5).

Exemple 4.1. (Stockage des opérateurs & Opérations algébriques en format séparé) Dans cet exemple, on décrit le stockage des opérateurs sous format séparé. On commente également le résultat obtenu suite à des opérations algébriques usuelles entre des objets donnés sous format séparé.

Stockage des opérateurs - Le format de stockage des opérateurs est schématisé sur la Figure 4.2. En pratique, l'implémentation est réalisée sous Matlab pour des systèmes linéaires à F champs dont les représentations discrètes sont des tenseurs d'ordre D. Dans ce cas, la représentation sous format séparé des opérateurs est donnée par

$$[\![\mathbf{A}]\!] \quad \text{avec} \quad \mathbf{A}_{ij} = \sum_{m=1}^{M_A(i,j)} \bigotimes_{d=1}^{D} \mathbf{A}_{ijm}^{d} \quad \text{et} \quad [\mathbf{a}] \quad \text{avec} \quad \mathbf{a}_i = \sum_{m=1}^{M_a(i)} \bigotimes_{d=1}^{D} \boldsymbol{a}_{im}^{d}, \tag{4.8}$$

où pour $i, j = 1, \cdots, F$, les composantes des opérateurs sont telles que $\mathbf{A}_{ijm}^{d} \in \mathbb{R}^{n_d(i)} \otimes \mathbb{R}^{n_d(j)}$ et $\boldsymbol{a}_{im}^{d} \in \mathbb{R}^{n_d(i)}$.

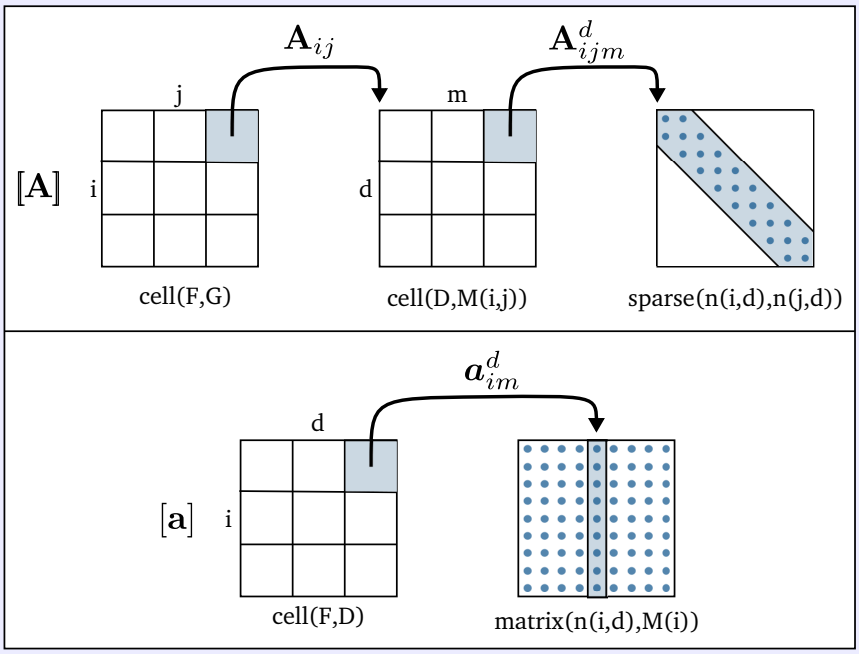

FIGURE 4.2: Implémentation des opérateurs sous format séparé (avec Matlab).

Opérations algébriques - Il est important de noter que les composantes $\mathbf{A}_{ij}$ et $\mathbf{a}_i$ des opérateurs $[\![\mathbf{A}]\!]$ et $[\mathbf{a}]$ appartiennent à des sous-ensembles de tenseurs qui ne forment pas des espaces vectoriels. Ces composantes appartiennent aux sous-ensembles des décompositions canoniques de rang $M_A(i, j)$ ou $M_b(i)$, respectivement (voir le récent ouvrage d'[Hackbusch, 2012] pour la théorie). Aussi, les opérations algébriques

entre ces opérateurs doivent être regardées avec précautions. En notant R_M le sous-ensemble des décompositions canoniques de rang M, on a notamment les résultats suivants sur les opérations courantes :

- étant donnés deux F-tuples $[\mathbf{a}]$ et $[\mathbf{b}]$, la somme

$$[\mathbf{a}] + [\mathbf{b}]\,, \tag{4.9}$$

  est un F-tuple $[\mathbf{c}]$ dont les composantes $\mathbf{c}_i$ appartiennent à $\mathrm{R}_{M_c(i)}$ avec

$$M_c(i) = M_a(i) + M_b(i) \quad \text{pour} \quad i = 1, \cdots, F\,, \tag{4.10}$$

- étant donnés un $(F \times G)$-tuple $[\![\mathbf{A}]\!]$ et un G-tuple $[\mathbf{a}]$, le produit

$$[\![\mathbf{A}]\!] \overset{\mathrm{D}}{\cdot} [\mathbf{a}]\,, \tag{4.11}$$

  est un F-tuple $[\mathbf{c}]$ dont les composantes $\mathbf{c}_i$ appartiennent à $\mathrm{R}_{M_c(i)}$ avec

$$M_c(i) = \sum_{j=1}^{G} M_A(i,j) M_a(j) \quad \text{pour} \quad i = 1, \cdots, F\,. \tag{4.12}$$

Aussi, on ne peut pas « écraser » le résultat d'une de ces opérations sur les cases mémoires utilisées pour stocker l'un des opérandes. Ceci peut notamment poser problème dans le cadre d'un processus itératif, où l'on souhaite par exemple mettre à jour un résidu sous la forme

$$[\mathbf{r}] = [\mathbf{r}] - [\![\mathbf{A}]\!] \overset{\mathrm{D}}{\cdot} [\mathbf{u}]\,. \tag{4.13}$$

Dans ce cas, l'espace mémoire utilisé pour stocker $[\mathbf{r}]$ augmentera à chaque itération (de même que la complexité de l'opération). Une solution peut être de tronquer le résidu courant à un certain rang [Ballani et Grasedyck, 2013].

4.2 Application à l'équation des ondes

Dans cette section, on détaille la construction des opérateurs **A** et **b** dans le cas d'une discrétisation espace-temps de l'équation des ondes. Une première illustration est donnée en partant du problème semi-discrétisé en espace puis approché en temps à l'aide d'un schéma d'intégration. La démarche est ensuite détaillée dans le cas où l'on connait une formulation faible du problème sur le domaine espace-temps. Des exemples à un champ et deux champs sont proposés. Enfin, pour chacun des cas, une attention particulière est portée à la prise en compte de conditions aux limites de types Dirichlet ou Neumann, et des conditions initiales en déplacement et vitesse.

Remarque 4.3. *Le problème de référence, ainsi que les notations utilisées dans cette section, sont détaillés dans le Chapitre 1.*

4.2.1 Construction à partir d'un schéma incrémental

Dans ce premier exemple, on montre comment construire la représentation tensorielle de opérateurs **A** et **b** dans le cas où le problème de référence est semi-discrétisé en espace par éléments finis, et approcher en temps à l'aide du schéma de Newmark. Dans ce cas, les opérateurs sont identifiés en écrivant l'ensemble des équations à tous les pickets de temps.

On suppose que le champ de déplacement a d'ores et déjà été approché en espace, sur une base de type éléments finis. Aussi, le point de départ est le problème semi-discrétisé en espace, dont on rappelle qu'il consiste à trouver le vecteur déplacement $\boldsymbol{U} : I \to \mathbb{R}^{n_S}$ qui vérifie

$$\mathbf{M}.\ddot{\boldsymbol{U}}(t) + \mathbf{K}.\boldsymbol{U}(t) = \boldsymbol{F}(t)\,, \tag{4.14a}$$

$$\text{avec} \quad \boldsymbol{U}(0) = \boldsymbol{U}_0\,, \tag{4.14b}$$

$$\text{et} \quad \dot{\boldsymbol{U}}(0) = \boldsymbol{V}_0\,, \tag{4.14c}$$

où **M** et **K** sont les matrices de masse et de raideur, $\boldsymbol{F}(t)$ est le vecteur des efforts extérieurs à un instant donné, et $\boldsymbol{U}_0$ et $\boldsymbol{V}_0$ sont les vecteurs déplacement initial et vitesse initiale.

Approximation temporelle avec le schéma de Newmark

Jusqu'ici la construction est tout à fait classique. En discrétisant le domaine temporel par $I = \{t_i \mid t_i = i\Delta t \text{ pour } i = 0, \cdots, n_T\}$ avec $t_0 = 0$ et $t_{n_T} = T$, puis en insérant les approximations de Newmark (1.22) dans l'équation de mouvement (4.14a) prise à l'instant t_i pour $i = 1, \cdots, n_T$, on obtient une formule de récurrence qui permet de résoudre le problème de façon incrémentale (voir le Schéma 1.1). La différence ici est que l'on écrit le schéma de Newmark comme un schéma à deux pas en privilégiant le déplacement. L'écriture du schéma sous cette forme est détaillée dans l'Exemple 4.2. Pour simplifier les notations, on note $\boldsymbol{U}(t_i) = \boldsymbol{U}_i$ et $\boldsymbol{F}(t_i) = \boldsymbol{F}_i$.

Définition 4.1. Le schéma de Newmark peut s'écrire sous la forme d'un schéma à deux pas, comme suit : calculer $\boldsymbol{U}_i$ pour $i = 2, \cdots, n_T$ en répétant :

$$\left(\tfrac{1}{\Delta t^2}\mathbf{M} + \beta\mathbf{K}\right).\boldsymbol{U}_i = b\,\boldsymbol{F}_{i-2} + a\,\boldsymbol{F}_{i-1} + \beta\,\boldsymbol{F}_i - \left(\tfrac{1}{\Delta t^2}\mathbf{M} + b\mathbf{K}\right).\boldsymbol{U}_{i-2} - \left(\tfrac{-2}{\Delta t^2}\mathbf{M} + a\mathbf{K}\right).\boldsymbol{U}_{i-1}\,,$$
$$\text{avec} \quad a = \tfrac{1}{2} - 2\beta + \gamma \quad \text{et} \quad b = \tfrac{1}{2} - \gamma + \beta\,. \tag{4.15}$$

Pour initialiser la formule de récurrence (4.15), on peut réaliser le premier incrément du schéma de Newmark écrit sous la forme du Schéma 1.1. On obtient pour $i = 1$,

$$\left(\tfrac{1}{\Delta t^2}\mathbf{M} + \beta\mathbf{K}\right).\boldsymbol{U}_1 = \beta\boldsymbol{F}_1 + \tfrac{1}{\Delta t^2}\mathbf{M}.\boldsymbol{U}_0 + \tfrac{1}{\Delta t}\mathbf{M}.\boldsymbol{V}_0 + (\tfrac{1}{2} - \beta)\mathbf{M}.\ddot{\boldsymbol{U}}_0\,, \tag{4.16}$$

où $\boldsymbol{U}_0$ et $\boldsymbol{V}_0$ sont les vecteurs déplacement initial et vitesse initiale, donnés par (4.14b) et (4.14c) (respectivement). On rappelle que l'accélération initiale $\ddot{\boldsymbol{U}}_0$ est donnée par l'équation de mouvement prise à l'instant $t = 0$, c'est-à-dire $\mathbf{M}.\ddot{\boldsymbol{U}}_0 = \boldsymbol{F}_0 - \mathbf{K}.\boldsymbol{U}_0$.

Remarque 4.4. *La forme à deux pas du schéma de Newmark est mentionnée à titre d'exemple dans [Hughes, 1987] à la page 527.*

Reformulation espace-temps sous format tensoriel

On reformule maintenant le problème incrémental sous la forme d'un unique système linéaire. Pour cela, on stocke le vecteur déplacement $\boldsymbol{U}_i$ à tous les intants t_i dans le tenseur du second ordre $\mathbf{U} \in \mathbb{R}^{n_S} \otimes \mathbb{R}^{n_T}$, comme suit :

$$\mathbf{U} = [\boldsymbol{U}_1, \cdots, \boldsymbol{U}_{n_T}]\,. \tag{4.17}$$

De la même façon, la i-ème colonne du tenseur $\mathbf{F} \in \mathbb{R}^{n_S} \otimes \mathbb{R}^{n_T}$ correspond au vecteur des efforts extérieurs $\boldsymbol{F}_i$ pris à l'instant t_i. On écrit ensuite la formule d'initialisation (4.16) et le schéma incrémental (4.15) à tous les instants pour obtenir un unique système d'équations. En regroupant les termes inconnus dans le membre de gauche et les termes connus dans le membre de droite, ce système d'équations s'écrit de la façon suivante :

$$\begin{cases} \left(\tfrac{1}{\Delta t^2}\mathbf{M} + \beta\mathbf{K}\right).\boldsymbol{U}_1 = \beta\boldsymbol{F}_1 + \tfrac{1}{\Delta t^2}\mathbf{M}.\boldsymbol{U}_0 + \tfrac{1}{\Delta t}\mathbf{M}.\boldsymbol{V}_0 + (\tfrac{1}{2} - \beta)\mathbf{M}.\ddot{\boldsymbol{U}}_0\,, \\ \left(\tfrac{-2}{\Delta t^2}\mathbf{M} + a\mathbf{K}\right).\boldsymbol{U}_1 + \left(\tfrac{1}{\Delta t^2}\mathbf{M} + \beta\mathbf{K}\right).\boldsymbol{U}_2 = a\boldsymbol{F}_1 + \beta\boldsymbol{F}_2 - \tfrac{1}{\Delta t^2}\mathbf{M}.\boldsymbol{U}_0 + b\mathbf{M}.\ddot{\boldsymbol{U}}_0\,, \\ \left(\tfrac{1}{\Delta t^2}\mathbf{M} + b\mathbf{K}\right).\boldsymbol{U}_1 + \left(\tfrac{-2}{\Delta t^2}\mathbf{M} + a\mathbf{K}\right).\boldsymbol{U}_2 + \left(\tfrac{1}{\Delta t^2}\mathbf{M} + \beta\mathbf{K}\right).\boldsymbol{U}_3 = b\boldsymbol{F}_1 + a\boldsymbol{F}_2 + \beta\boldsymbol{F}_3\,, \\ \vdots \\ \left(\tfrac{1}{\Delta t^2}\mathbf{M} + b\mathbf{K}\right).\boldsymbol{U}_{n_T-2} + \left(\tfrac{-2}{\Delta t^2}\mathbf{M} + a\mathbf{K}\right).\boldsymbol{U}_{n_T-1} + \left(\tfrac{1}{\Delta t^2}\mathbf{M} + \beta\mathbf{K}\right).\boldsymbol{U}_{n_T} = \\ b\boldsymbol{F}_{n_T-2} + a\boldsymbol{F}_{n_T-1} + \beta\boldsymbol{F}_{n_T}\,. \end{cases} \tag{4.18}$$

Le système (4.18) regroupe l'ensemble des équations à résoudre, pour obtenir l'approximation de Newmark de la solution (du problème (4.14a)) sur tout le domaine espace-temps, c'est-à-dire pour obtenir le tenseur $\mathbf{U}$. Pour écrire ce système sous format tensoriel, il suffit d'identifier les opérateurs $\mathbf{A}_k^S$, $\mathbf{A}_k^T$, $\boldsymbol{b}_k^S$ et $\boldsymbol{b}_k^T$ de l'équation (4.2). En remarquant que le système linéaire $(\mathbf{A}^S \otimes \mathbf{A}^T) \overset{\mathrm{D}}{\cdot} \mathbf{U} = \boldsymbol{b}^S \otimes \boldsymbol{b}^T$ peut s'implémenter sous la forme [Van Loan, 2000],

$$\begin{bmatrix} \mathrm{A}_{11}^T\mathbf{A}^S & \cdots & \mathrm{A}_{1n_T}^T\mathbf{A}^S \\ | & \diagdown & | \\ \mathrm{A}_{n_T1}^T\mathbf{A}^S & \cdots & \mathrm{A}_{n_Tn_T}^T\mathbf{A}^S \end{bmatrix} . \begin{bmatrix} \boldsymbol{U}_1 \\ | \\ \boldsymbol{U}_{n_T} \end{bmatrix} = \begin{bmatrix} b_1^T\boldsymbol{b}^S \\ | \\ b_{n_T}^T\boldsymbol{b}^S \end{bmatrix}, \tag{4.19}$$

et en utilisant la propriété suivante,

$$\left(\mathbf{A}_1^S \otimes \mathbf{A}_1^T + \mathbf{A}_2^S \otimes \mathbf{A}_2^T\right) \overset{\mathrm{D}}{\cdot} \mathbf{U} = \left(\mathbf{A}_1^S \otimes \mathbf{A}_1^T\right) \overset{\mathrm{D}}{\cdot} \mathbf{U} + \left(\mathbf{A}_2^S \otimes \mathbf{A}_2^T\right) \overset{\mathrm{D}}{\cdot} \mathbf{U}, \tag{4.20}$$

on peut facilement identifier les opérateurs de Newmark associés à la discrétisation du problème sur tout l'intervalle de temps.

Problème 4.1. Le problème espace-temps associé à une discrétisation en temps avec le schéma de Newmark, consiste à trouver $\mathbf{U} \in \mathbb{R}^{n_S} \otimes \mathbb{R}^{n_T}$ tel que

$$\begin{aligned}\left(\mathbf{K} \otimes \mathbf{N}_1 + \mathbf{M} \otimes \mathbf{N}_2\right) \overset{\mathrm{D}}{\cdot} \mathbf{U} = (\mathbf{I} \otimes \mathbf{N}_1) \overset{\mathrm{D}}{\cdot} \mathbf{F} \\ + (\mathbf{M}.\boldsymbol{U}_0) \otimes \boldsymbol{N}_3 \\ + (\mathbf{M}.\boldsymbol{V}_0) \otimes \boldsymbol{N}_4 \\ + (\boldsymbol{F}_0 - \mathbf{K}.\boldsymbol{U}_0) \otimes \boldsymbol{N}_5,\end{aligned} \tag{4.21}$$

avec les définitions suivantes des opérateurs en temps associés aux approximations de Newmark :

$$\mathbf{N}_1 = \begin{bmatrix} \beta & 0 & \cdots & 0 \\ a & \ddots & 0 & \vdots \\ b & \ddots & \ddots & 0 \\ 0 & b & a & \beta \end{bmatrix}, \quad \mathbf{N}_2 = \tfrac{1}{\Delta t^2} \begin{bmatrix} 1 & 0 & \cdots & 0 \\ -2 & \ddots & 0 & \vdots \\ 1 & \ddots & \ddots & 0 \\ 0 & 1 & -2 & 1 \end{bmatrix}$$

$$\boldsymbol{N}_3 = \tfrac{1}{\Delta t^2} \begin{bmatrix} 1 \\ -1 \\ 0 \\ \vdots \end{bmatrix}, \quad \boldsymbol{N}_4 = \tfrac{1}{\Delta t} \begin{bmatrix} 1 \\ 0 \\ \vdots \end{bmatrix} \text{ et } \quad \boldsymbol{N}_5 = \begin{bmatrix} \frac{1}{2} - \beta \\ b \\ 0 \\ \vdots \end{bmatrix},$$

où $a = \frac{1}{2} - 2\beta + \gamma$ et $b = \frac{1}{2} - \gamma + \beta$.

Remarque 4.5. *Dans la publication [Boucinha* et al.*, 2013b], on a identifié les (mêmes) opérateurs en temps en partant des approximations de Newmark sous la forme (1.22), et en écrivant le problème espace-temps comme un problème à trois champs (déplacement, vitesse, accélération).*

Remarque 4.6. *La formalisation du problème sur tous les pickets de temps a également été considérée par [Gravouil, 2000] en prévilégiant le vecteur accélération dans les approximations de Newmark.*

Remarque 4.7. *En utilisant le produit de Kronecker [Van Loan, 2000], le système (4.21) peut être implémenté sous la forme d'un système linéaire $\mathbf{A}.\boldsymbol{u} = \boldsymbol{b}$ avec $\mathbf{A} \in \mathbb{R}^n \otimes \mathbb{R}^n$ et $\boldsymbol{u}, \boldsymbol{b} \in \mathbb{R}^n$ où $n = n_S n_T$. On obtient alors la même solution (aux erreurs d'arrondis près) en*

résolvant ce système linéaire à l'aide d'un solveur classique d'algèbre linéaire ou en utilisant le schéma incrémental. Cependant, une résolution brutale de ce système linéaire est bien plus coûteuse qu'une résolution incrémentale. Très grossièrement[11]*, si* $\boldsymbol{lin}(n)$ *est la complexité associée à la résolution d'un système linéaire de taille* $n \times n$*, il faut* $\boldsymbol{lin}(n_S n_T)$ *opérations pour résoudre le problème espace-temps avec un solveur classique d'algèbre linéaire alors que seulement* $n_T \boldsymbol{lin}(n_S)$ *opérations sont nécessaires avec le schéma incrémental.*

Conditions aux limites et initiales

En regardant le second membre de l'équation (4.21), on peut voir que les conditions initiales en déplacement et vitesse sont imposées naturellement, avec la procédure d'initialisation classique associée au schéma de Newmark. On notera également que ces conditions sont imposées de façon forte [12].

Dans ce second membre, on a volontairement écrit le tenseur $\mathbf{F}$ sous format non-séparé. Ce tenseur est associé aux efforts extérieurs et peut être séparé comme suit :

$$\mathbf{F} = \mathbf{F}_p - (\mathbf{F}_g + \mathbf{F}_{\ddot{g}}), \tag{4.22}$$

où $\mathbf{F}_p$ est la contribution due à l'effort ponctuel $p(t)$ imposé au point $x \in \partial\Omega_\sigma$, et $\mathbf{F}_g + \mathbf{F}_{\ddot{g}}$ sont les contributions dues au déplacement $g(t)$ imposé au point $x \in \partial\Omega_u$. On regardant l'équation (1.13), on peut exprimer ces tenseurs sous format séparé de la façon suivante :

$$\boxed{\mathbf{F}_p = \phi|_{\Omega_\sigma} \otimes \boldsymbol{p}, \quad \mathbf{F}_g = \boldsymbol{K}_g \otimes \boldsymbol{g}, \quad \text{et} \quad \mathbf{F}_{\ddot{g}} = \boldsymbol{M}_g \otimes \ddot{\boldsymbol{g}}}, \tag{4.23a}$$

$$\text{avec} \quad \boldsymbol{K}_g = k(\phi, \phi^g), \quad \boldsymbol{M}_g = m(\phi, \phi^g) \tag{4.23b}$$

$$\text{et} \quad \boldsymbol{p} = \begin{bmatrix} p(t_1) \\ | \\ p(t_{n_T}) \end{bmatrix}, \quad \boldsymbol{g} = \begin{bmatrix} g(t_1) \\ | \\ g(t_{n_T}) \end{bmatrix}, \quad \ddot{\boldsymbol{g}} = \begin{bmatrix} \ddot{g}(t_1) \\ | \\ \ddot{g}(t_{n_T}) \end{bmatrix}. \tag{4.23c}$$

où on rappelle que ϕ est la base éléments finis de $\mathscr{U}_h^S(\Omega;0)$ et ϕ^g est la fonction de forme associée au singleton $\partial\Omega_u$.

Remarque 4.8. *Dans le cas général, si une fonction* $f(x,t)$ *associée aux conditions aux limites, est connue sous la forme* $\sum_{k=1}^{M_f} f_k^S(x) f_k^T(t)$*, alors le tenseur* $\mathbf{F}$ *peut aisément se décomposer sous la forme* $\sum_{k=1}^{M_f} \boldsymbol{F}_k^S \otimes \boldsymbol{F}_k^T$*. Si cette fonction n'est pas connue explicitement dans cette forme (par exemple dans le cas d'une charge mobile), on peut toujours calculer une meilleure approximation de celle-ci dans* $\mathscr{R}_{M_f}$ *(en adaptant* M_f *pour que l'approximation soit suffisamment précise).*

11. Dans cette évaluation, on ne tient pas compte de la largeur de bande de la matrice considérée, ni de son conditionnement, qui de toute façon avantagent le schéma incrémental.

12. On impose a priori que le vecteur déplacement, et ses dérivées en temps vérifient les conditions initiales, c'est-à-dire que l'instant initial ne fait pas parti du domaine de calcul.

Exemple 4.2. (Schéma de Newmark à deux pas) Dans cet exemple, on montre comment écrire le schéma de Newmark sous la forme d'un schéma incrémental à deux pas en déplacement. Les étapes du calcul sont les suivantes :

1. L'objectif est d'obtenir une formule qui permette de relier : le vecteur déplacement $\boldsymbol{U}_{i+1}$ vérifiant l'équation de mouvement à l'instant t_{i+1}, avec les vecteurs déplacements $\boldsymbol{U}_i$ et $\boldsymbol{U}_{i-1}$ vérifiant l'équation de mouvement aux instants t_i et t_{i-1}. Pour cela, on écrit tout d'abord l'équation de mouvement (4.14a) aux instants t_{i-1}, t_i et t_{i+1}, soit :

$$\mathbf{M}.\ddot{\boldsymbol{U}}_{i-1} + \mathbf{K}.\boldsymbol{U}_{i-1} = \boldsymbol{F}_{i-1}\,, \tag{4.24a}$$

$$\mathbf{M}.\ddot{\boldsymbol{U}}_{i} + \mathbf{K}.\boldsymbol{U}_{i} = \boldsymbol{F}_{i}\,, \tag{4.24b}$$

$$\mathbf{M}.\ddot{\boldsymbol{U}}_{i+1} + \mathbf{K}.\boldsymbol{U}_{i+1} = \boldsymbol{F}_{i+1}\,. \tag{4.24c}$$

2. Pour relier les différentes quantités entre deux instants, on utilise les approximations de Newmark (1.22), que l'on écrit ici à l'instant t_{i+1}, en privilégiant le vecteur déplacement, c'est-à-dire

$$\dot{\boldsymbol{U}}_{i+1} = \tfrac{\gamma}{\beta\Delta t}\big(\boldsymbol{U}_{i+1} - \boldsymbol{U}_i\big) + (1-\tfrac{\gamma}{\beta})\dot{\boldsymbol{U}}_i + \Delta t(1-\tfrac{\gamma}{2\beta})\ddot{\boldsymbol{U}}_i\,, \tag{4.24d}$$

$$\ddot{\boldsymbol{U}}_{i+1} = \tfrac{1}{\beta\Delta t^2}\big(\boldsymbol{U}_{i+1} - \boldsymbol{U}_i\big) - \tfrac{1}{\beta\Delta t}\dot{\boldsymbol{U}}_i + (1-\tfrac{1}{2\beta})\ddot{\boldsymbol{U}}_i\,. \tag{4.24e}$$

3. On relit dans un premier temps les équations de mouvement prises aux instants t_{i+1} et t_i. Pour cela, on remplace l'approximation de Newmark de $\ddot{\boldsymbol{U}}_{i+1}$ donnée par l'équation (4.24e) dans l'équation de mouvement (4.24c). On obtient l'expression suivante,

$$\big(\tfrac{1}{\Delta t^2}\mathbf{M} + \beta\mathbf{K}\big).\boldsymbol{U}_{i+1} = \beta\boldsymbol{F}_{i+1} + \tfrac{1}{\Delta t^2}\mathbf{M}.\boldsymbol{U}_i + \tfrac{1}{\Delta t}\mathbf{M}.\dot{\boldsymbol{U}}_i + (\tfrac{1}{2}-\beta)\mathbf{M}.\ddot{\boldsymbol{U}}_i\,, \tag{4.24f}$$

qui peut s'écrire, en remarquant que $\mathbf{M}.\ddot{\boldsymbol{U}}_i = \boldsymbol{F}_i - \mathbf{K}.\boldsymbol{U}_i$ d'après (4.24b), sous la forme :

$$\big(\tfrac{1}{\Delta t^2}\mathbf{M} + \beta\mathbf{K}\big).\boldsymbol{U}_{i+1} = \beta\boldsymbol{F}_{i+1} + (\tfrac{1}{2}-\beta)\boldsymbol{F}_i + \big(\tfrac{1}{\Delta t^2}\mathbf{M} + (\beta-\tfrac{1}{2})\mathbf{K}\big).\boldsymbol{U}_i + \tfrac{1}{\Delta t}\mathbf{M}.\dot{\boldsymbol{U}}_i\,. \tag{4.24g}$$

4. On établit maintenant le lien entre les équations de mouvements prises aux instants t_i et t_{i-1}. En répétant l'étape précédente en remplaçant t_{i+1} par t_i et t_i par t_{i-1}, on obtient :

$$\big(\tfrac{1}{\Delta t^2}\mathbf{M} + \beta\mathbf{K}\big).\boldsymbol{U}_{i} = \beta\boldsymbol{F}_{i} + (\tfrac{1}{2}-\beta)\boldsymbol{F}_{i-1} + \big(\tfrac{1}{\Delta t^2}\mathbf{M} + (\beta-\tfrac{1}{2})\mathbf{K}\big).\boldsymbol{U}_{i-1} + \tfrac{1}{\Delta t}\mathbf{M}.\dot{\boldsymbol{U}}_{i-1}\,. \tag{4.24h}$$

5. Il reste alors à relier les équations (4.24g) et (4.24h) entre elles. Pour cela, on utilise l'approximation de Newmark de $\dot{\boldsymbol{U}}_i$ donnée par l'équation (4.24d) prise à l'instant t_i. En multipliant cette équation par $\frac{1}{\Delta t}\mathbf{M}$, on a :

$$\tfrac{1}{\Delta t}\mathbf{M}.\dot{\boldsymbol{U}}_i = \tfrac{\gamma}{\beta\Delta t^2}\mathbf{M}.\big(\boldsymbol{U}_i - \boldsymbol{U}_{i-1}\big) + (\tfrac{1}{\Delta t} - \tfrac{\gamma}{\beta\Delta t})\mathbf{M}.\dot{\boldsymbol{U}}_{i-1} + (1 - \tfrac{\gamma}{2\beta})\mathbf{M}.\ddot{\boldsymbol{U}}_{i-1}\,. \tag{4.24i}$$

En remplaçant alors l'expression de $\frac{1}{\Delta t}\mathbf{M}.\dot{\boldsymbol{U}}_{i-1}$ donnée par (4.24h) dans l'équation (4.24i), et en remarquant que $\mathbf{M}.\ddot{\boldsymbol{U}}_{i-1} = \boldsymbol{F}_{i-1} - \mathbf{K}.\boldsymbol{U}_{i-1}$ d'après (4.24a), on obtient l'expression suivante de $\frac{1}{\Delta t}\mathbf{M}.\dot{\boldsymbol{U}}_i$:

$$\begin{aligned}&\tfrac{1}{\Delta t}\mathbf{M}.\dot{\boldsymbol{U}}_i = \big(\tfrac{1}{\Delta t^2}\mathbf{M} + (\beta-\gamma)\mathbf{K}\big).\boldsymbol{U}_i - \big(\tfrac{1}{\Delta t^2}\mathbf{M} + b\mathbf{K}\big).\boldsymbol{U}_{i-1} + (\gamma-\beta)\boldsymbol{F}_i + b\boldsymbol{F}_{i-1}\,,\\ &\text{avec}\quad b = \tfrac{1}{2} - \gamma + \beta\,.\end{aligned} \tag{4.24j}$$

6. Finalement, en remplaçant l'expression de $\frac{1}{\Delta t}\mathbf{M}.\dot{\boldsymbol{U}}_i$ donnée par (4.24j) dans l'équation (4.24g), on obtient l'algorithme à deux pas du schéma de Newmark écrit en déplacement :

$$\begin{aligned}&\big(\tfrac{1}{\Delta t^2}\mathbf{M} + \beta\mathbf{K}\big).\boldsymbol{U}_{i+1} = b\,\boldsymbol{F}_{i-1} + a\,\boldsymbol{F}_i + \beta\,\boldsymbol{F}_{i+1} - \big(\tfrac{1}{\Delta t^2}\mathbf{M} + b\mathbf{K}\big).\boldsymbol{U}_{i-1} - \big(\tfrac{-2}{\Delta t^2}\mathbf{M} + a\mathbf{K}\big).\boldsymbol{U}_i\,,\\ &\text{avec}\quad a = \tfrac{1}{2} - 2\beta + \gamma\,.\end{aligned} \tag{4.24k}$$

4.2.2 Construction avec une formulation faible espace-temps

Dans ce deuxième exemple, on montre comment construire les opérateurs **A** et **b** lorsque l'on connait une formulation faible du problème sur le domaine espace-temps. Dans ce cas, le formalisme tensoriel peut être introduit dès l'étape de discrétisation espace-temps et l'identification des opérateurs est simplifiée. La stratégie est esquissée dans un cadre général pour une formulation à F champs. Les opérateurs obtenus en appliquant cette stratégie dans le cas des méthodes éléments finis en temps présentées dans le Chapitre 1 sont ensuite précisés. Des détails supplémentaires peuvent être trouvés dans [Boucinha *et al.*, 2013b].

Méthode éléments finis espace-temps à F champs

Dans un cadre général d'un problème à F-champs, une formulation faible espace-temps consiste à trouver les champs $u_j \in \mathscr{U}_j(\Omega \times I)$ pour $j = 1, \cdots, F$ qui vérifient

$$\sum_{i=1}^{F}\sum_{j=1}^{F} a_{ij}(u_i^\star, u_j) = \sum_{i=1}^{F} b_i(u_i^\star)\,, \quad \forall\, u_i^\star \in \mathscr{U}_i(\Omega \times I) \text{ pour } i = 1, \cdots, F\,, \tag{4.25}$$

où les termes $a_{ij}(\,,\,)$ sont des formes bilinéaires définies sur $\mathscr{U}_i \times \mathscr{U}_j$ et les termes $b_i()$ sont des formes linéaires définies sur $\mathscr{U}_i$. On suppose maintenant que tous les espaces

$\mathscr{U}_i(\Omega \times I)$ sont des espaces produits tensoriels, c'est-à-dire $\mathscr{U}_i(\Omega \times I) = \mathscr{U}_i^S(\Omega) \otimes \mathscr{U}_i^T(I)$. Alors, les composantes des opérateurs du problème espace-temps peuvent être identifiées directement dans tous les cas où les formes $a_{ij}(,)$ et $b_i()$ vérifient les propriétés de séparabilité suivantes $\forall w_i^\star \in \mathscr{U}_i^S, \forall w_j \in \mathscr{U}_j^S$ et $\forall \lambda_i^\star \in \mathscr{U}_i^T, \forall \lambda_j \in \mathscr{U}_j^T$:

$$a_{ij}(w_i^\star \lambda_i^\star, w_j \lambda_j) = \sum_{k=1}^{M_A(i,j)} a_{ijk}^S(w_i^\star, w_j) a_{ijk}^T(\lambda_i^\star, \lambda_j), \tag{4.26a}$$

$$\text{et} \quad b_i(w_i^\star \lambda_i^\star) = \sum_{k=1}^{M_b(i)} b_{ik}^S(w_i^\star) b_{jk}^T(\lambda_i^\star), \tag{4.26b}$$

où les termes $a_{ijk}^S(,)$ et $a_{ijk}^T(,)$ désignent des formes bilinéaires définies respectivement sur $\mathscr{U}_i^S \times \mathscr{U}_j^S$ et $\mathscr{U}_i^T \times \mathscr{U}_j^T$, et les termes $b_{ik}^S()$ et $b_{ik}^T()$ sont des formes linéaires définies respectivement sur $\mathscr{U}_i^S$ et $\mathscr{U}_i^T$. La méthode des éléments finis espace-temps consiste alors à remplacer les espaces $\mathscr{U}_i^S \otimes \mathscr{U}_i^T$ par des espaces de dimension finie [13] (noté $\mathscr{U}_{h,i}^S \otimes \mathscr{U}_{\Delta t,i}^T$) construit par tensorisation de bases éléments finis en espace (noté ϕ_i) et en temps (noté ψ_i). En écrivant alors la formulation faible sur les espaces $\mathscr{U}_{h,i}^S \otimes \mathscr{U}_{\Delta t,i}^T$ et en remarquant qu'elle est vérifiée quelles que soient les fonctions tests, on aboutit finalement à un système linéaire de la forme $[\![\mathbf{A}]\!] \overset{\mathrm{D}}{\cdot} [\mathbf{u}] = [\mathbf{b}]$, dont les composantes des opérateurs (donnés sous la forme de l'équation (4.4)) peuvent être identifiées par

$$\boxed{\begin{array}{ll} & \mathbf{A}_{ijk}^S = a_{ijk}^S(\phi_i, \phi_j), \quad \mathbf{A}_{ijk}^T = a_{ijk}^T(\psi_i, \psi_j), \quad \text{pour } k = 1, \cdots, M_A(i,j), \\ \text{et} & \boldsymbol{b}_{ik}^S = b_{ik}^S(\phi_i), \qquad \boldsymbol{b}_{ik}^T = b_{ik}^T(\phi_i), \qquad \text{pour } k = 1, \cdots, M_b(i). \end{array}} \tag{4.27}$$

Application aux méthodes éléments finis en temps du Chapitre 1

Les méthodes éléments finis en temps présentées au Chapitre 1 peuvent être écrites sous la forme de l'équation (4.25) en partant du problème continu en espace [14] [15]. L'identification des termes de l'équation (4.26a) ne pose pas de problème technique particulier. Les opérateurs obtenus pour les trois méthodes présentées au Chapitre 1 (à savoir TDG-U, TDG-UV et TG-UV) sont regroupés dans le Tableau 4.1. Tous les détails concernant leur identifiaction peuvent être trouvés dans [Boucinha *et al.*, 2013b].

L'identification des termes de l'équation (4.26b) est un peu plus technique. Le point clé concerne le choix des espaces pour imposer les conditions aux limites (Dirichlet

13. On autorise des espaces d'approximation de tailles différentes pour chaque champ.

14. La formulation originale de [Hulbert et Hughes, 1990] est une formulation faible espace-temps, écrite par tranche de temps. Il suffit d'écrire cette formulation sur toutes les tranches de temps pour se mettre sous la forme générale de l'équation (4.25).

15. Pour la méthode de Galerkin continue en temps, il suffit d'écrire l'équation (1.34a) en partant du problème continu en espace

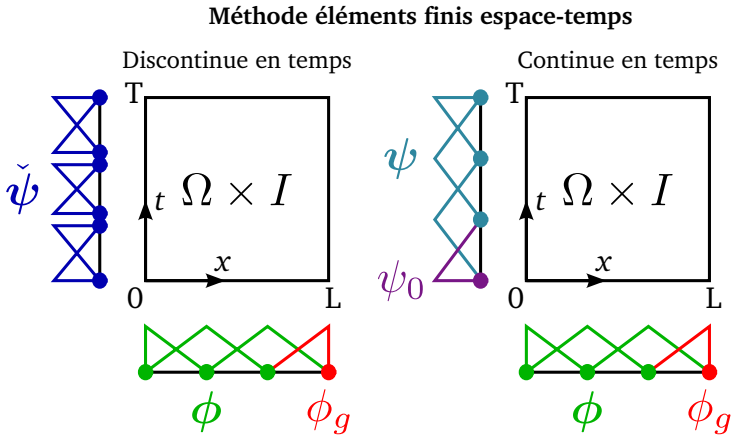

FIGURE 4.3: Base éléments finis espace-temps pour différentes méthodes.

et Neumann) et les conditions initiales (en déplacement et vitesse). On précise ici ce choix pour les trois méthodes éléments finis en temps du Chapitre 1. La Figure 4.3 pourra aider à la compréhension.

- **(TDG-U)** Pour la méthode de Galerkin discontinue en temps, les conditions initiales sont imposées de façon faible dans la formulation. Aussi, on doit seulement imposer la condition de Dirichlet dans l'espace fonctionnel, c'est-à-dire que le champ de déplacement est cherché dans l'espace

$$\mathscr{U}(\Omega\times I;g)=\left\{u\in\mathscr{U}(\Omega\times I)\mid u(x,t)=g(x,t)\quad\forall(x,t)\in\partial\Omega_u\times I\right\}. \tag{4.28}$$

  En suivant la Remarque 1.4 et en introduisant les espaces d'approximation éléments finis $\mathscr{U}_h^S(\Omega;0)$ et $\mathscr{U}_{\Delta t}^{\check{T}}(I)$ (respectivement continus en espace et discontinus en temps), on approche alors le champ de déplacement $u\in\mathscr{U}(\Omega\times I;g)$ sous la forme

$$u\simeq u_{h,\Delta t}+\check{g}\quad\text{avec}\quad\begin{cases}u_{h,\Delta t}\in\mathscr{U}_h^S(\Omega;0)\otimes\mathscr{U}_{\Delta t}^{\check{T}}(I)\\ \check{g}(x,t)=g(x,t)\quad\forall(x,t)\in\partial\Omega_u\times I\end{cases}. \tag{4.29}$$

  En introduisant les bases éléments finis ϕ de $\mathscr{U}_h^S(\Omega;0)$ et $\check{\psi}$ de $\mathscr{U}_{\Delta t}^{\check{T}}(I)$, ainsi que les fonctions de forme ϕ_g décritent dans Exemple 1.2, cette approximation s'écrit également [16]

$$u(x,t)\simeq\phi(x)\otimes\check{\psi}(t)\,\underline{\underline{\mathrm{D}}}\,\check{\mathbf{U}}+\phi_g(x)\otimes\check{\psi}(t)\,\underline{\underline{\mathrm{D}}}\,\check{\mathbf{g}}. \tag{4.30}$$

  Finalement, la fonction test $u^\star$ est prise dans $\mathscr{U}_h^S(\Omega;0)\otimes\mathscr{U}_{\Delta t}^{\check{T}}(I)$, c'est-à-dire

$$u^\star(x,t)=\phi(x)\otimes\check{\psi}(t)\,\underline{\underline{\mathrm{D}}}\,\check{\mathbf{U}}^\star. \tag{4.31}$$

16. Lorsque $\partial\Omega_u$ est un singleton, on a directement $\check{g}(x,t)=\phi_g(x)\otimes(\check{\psi}(t).\check{\mathbf{g}})$ avec $g(t)\simeq(\check{\psi}(t).\check{\mathbf{g}})$.

Il reste alors à discrétiser d'une part, le déplacement initial $u_0(x)$ et la vitesse initiale $v_0(x)$ (on choisit simplement [17] $u_0(x) \simeq \phi(x).\boldsymbol{U}_0$ et $v_0(x) \simeq \phi(x).\boldsymbol{V}_0$) et d'autre part, le chargement extérieur $p(t)$ (on choisit $p(t) \simeq \check{\psi}(t).\check{\boldsymbol{p}}$).

- **(TDG-UV)** Pour la méthode de Galerkin discontinue en temps à deux champs (déplacement-vitesse), le choix des espaces d'approximation est très similaire à celui de la méthode TDG-U. On notera cependant que le champ de vitesse est ici un champ à part entière. Aussi, on impose également une condition de Dirichlet en vitesse dans l'espace fonctionnel, c'est-à-dire que le champ de vitesse est cherché dans l'espace

$$\mathcal{V}(\Omega \times I; \dot{g}) = \left\{ v \in \mathcal{V}(\Omega \times I) \mid v(x,t) = \dot{g}(x,t) \quad \forall (x,t) \in \partial\Omega_u \times I \right\}, \tag{4.32}$$

avec $\dot{g} = \frac{\mathrm{d}g}{\mathrm{d}t}$. Le choix des espaces d'approximation pour les champs de déplacement et de vitesse est ensuite similaire [18] à celui de la méthode TDG-U.

- **(TG-UV)** Dans le cas de la méthode de Galerkin continue en temps, on doit imposer les conditions initiales, en plus de la condition de Dirichlet, dans l'espace fonctionnel [19]. On détaille cet aspect pour le champ de déplacement, la construction étant similaire pour le champ de vitesse. Aussi, on cherche le champ de déplacement dans l'espace

$$\mathcal{U}(\Omega \times I; g, u_0) = \left\{ u \in \mathcal{U}(\Omega \times I; g) \mid u(x,0) = u_0(x) \quad \forall x \in \Omega \right\}. \tag{4.33}$$

En suivant de nouveau la Remarque 1.4 et en introduisant les espaces d'approximation éléments finis $\mathcal{U}_h^S(\Omega;0)$ et $\mathcal{U}_{\Delta t}^T(I;0)$ (respectivement continus en espace et en temps), on approche alors le champ de déplacement $u \in \mathcal{U}(\Omega \times I; g, u_0)$ sous la forme

$$u \simeq u_{h,\Delta t} + \tilde{g} + \tilde{u}_0 \quad \text{avec} \quad \begin{cases} u_{h,\Delta t} \in \mathcal{U}_h^S(\Omega;0) \otimes \mathcal{U}_{\Delta t}^T(I;0) \\ \tilde{g}(x,t) = g(x,t) \quad \forall (x,t) \in \partial\Omega_u \times I \\ \tilde{u}_0(x,0) = u_0(x) \quad \forall x \in \Omega \end{cases}. \tag{4.34}$$

En introduisant les bases éléments finis ϕ de $\mathcal{U}_h^S(\Omega;0)$ et ψ de $\mathcal{U}_{\Delta t}^T(I;0)$, ainsi que les fonctions de forme ϕ_g et ψ_0 représentées sur la Figure 4.3, cette approximation s'écrit également [20]

$$u(x,t) \simeq \phi(x) \otimes \psi(t) \overset{\mathrm{D}}{\cdot} \mathbf{U} + \phi_g(x) \otimes \psi(t) \overset{\mathrm{D}}{\cdot} \mathbf{g} + (\phi(x).\boldsymbol{U}_0)\psi_0(t). \tag{4.35}$$

17. Pour être précis, on doit écrire $u_0 \in \mathcal{U}^S(\Omega)$ sous la forme $u_0(x) \simeq \phi(x).\boldsymbol{U}_0 + \phi_g(x)u_0^g$. Mais le terme $\phi_g(x)u_0^g$ s'annulera dans la formulation avec le terme $\check{g}(x,0)$ si la mise en données vérifie $u_0(x) = g(x,0)$, $\forall x \in \partial\Omega_u$. Et de même, pour la vitesse initiale.

18. La formulation à deux champs impose la continuité entre le champ de vitesse et la dérivée temporelle du champ de déplacement de façon faible. Aussi, lorsque la mise en donnée vérifie exactement $\frac{\mathrm{d}g}{\mathrm{d}t} = \dot{g}$, alors les termes permettant d'imposer la continuité entre $\frac{\mathrm{d}\check{g}}{\mathrm{d}t}$ et $\dot{\check{g}}$ (liés à l'écriture du champ de déplacement et de vitesse sous la forme de l'équation (4.29)) s'annulent.

19. Dans [Boucinha *et al.*, 2013b], les conditions initiales sont imposées de façon faible pour la méthode TG-UV.

20. Pour être précis, on devrait également ajouter un terme $(\phi_g(x).\boldsymbol{g}_0)\psi_0(t)$ (associé à $g(x,0)$ pour $x \in \partial\Omega_u$). On suppose ici que ce terme est nul (c'est-à-dire que $g(x,0) = 0$ pour $x \in \partial\Omega_u$).

On prend finalement la fonction test $u^\star$ dans $\mathscr{U}_h^S(\Omega;0) \otimes \mathscr{U}_{\Delta t}^T(I;0)$, c'est-à-dire

$$u^\star(x,t) = \phi(x) \otimes \psi(t) \overset{\mathrm{D}}{\cdot} \mathbf{U}^\star . \tag{4.36}$$

Il reste alors à discrétiser le chargement extérieur $p(t)$ sur I, on choisit $p(t) \simeq \psi(t).\boldsymbol{p} + \psi_0(t)p_0$.

Les opérateurs de l'équation (4.26b) obtenus de cette façon pour les trois méthodes, sont regroupés dans le Tableau 4.2. Afin de simplifier les notations, les opérateurs en temps sont donnés, pour les formulation à deux champs (déplacement-vitesse), dans le cas où les mêmes espaces d'approximation ont été utilisés pour discrétiser ces champs.

Méthode	$[\![\mathbf{A}]\!]$		$[\mathbf{u}]$	Définition des opérateurs en temps
Newmark	$\mathbf{K}\otimes\mathbf{N}_1+\mathbf{M}\otimes\mathbf{N}_2$		$\mathbf{U}$	$\mathbf{N}_1=\begin{bmatrix}\beta & 0 & - & 0\\ a & \diagdown & 0 & \vert\\ b & \diagdown & \diagdown & 0\\ 0 & b & a & \beta\end{bmatrix}$ $\mathbf{N}_2=\frac{1}{\Delta t^2}\begin{bmatrix}1 & 0 & - & 0\\ -2 & \diagdown & 0 & \vert\\ 1 & \diagdown & \diagdown & 0\\ 0 & 1 & -2 & 1\end{bmatrix}$ $\boldsymbol{N}_3=\frac{1}{\Delta t^2}\begin{bmatrix}1\\ -1\\ 0\\ \vert\end{bmatrix}$ $\boldsymbol{N}_4=\frac{1}{\Delta t}\begin{bmatrix}1\\ 0\\ \vert\end{bmatrix}$ $\boldsymbol{N}_5=\begin{bmatrix}\frac{1}{2}-\beta\\ b\\ 0\\ \vert\end{bmatrix}$
TDG-U	$\mathbf{K}\otimes(\check{\mathbf{Q}}^{10}+\check{\mathbf{P}}^{00})+\mathbf{M}\otimes(\check{\mathbf{Q}}^{12}+\check{\mathbf{P}}^{11})$		$\check{\mathbf{U}}$	$\check{\mathbf{Q}}^{kl}=\sum_{i=1}^{N_T}\int_{I_i}\frac{\mathrm{d}^k\check{\psi}}{\mathrm{d}t^k}\otimes\frac{\mathrm{d}^l\check{\psi}}{\mathrm{d}t^l}\,\mathrm{dt}$
TDG-UV	$\mathbf{K}\otimes(\check{\mathbf{Q}}^{01}+\check{\mathbf{P}}^{00})$	$-\mathbf{K}\otimes\check{\mathbf{Q}}^{00}$	$\check{\mathbf{U}}$	$\check{\mathbf{P}}^{kl}=\frac{\mathrm{d}^k\check{\psi}(t_0^+)}{\mathrm{d}t^k}\otimes\frac{\mathrm{d}^l\check{\psi}(t_0^+)}{\mathrm{d}t^l}+\sum_{i=2}^{N_T}\left(\frac{\mathrm{d}^k\check{\psi}(t_{i-1}^+)}{\mathrm{d}t^k}\otimes\frac{\mathrm{d}^l\check{\psi}(t_{i-1}^+)}{\mathrm{d}t^l}-\frac{\mathrm{d}^k\check{\psi}(t_{i-1}^+)}{\mathrm{d}t^k}\otimes\frac{\mathrm{d}^l\check{\psi}(t_{i-1}^-)}{\mathrm{d}t^l}\right)$
	$\mathbf{K}\otimes\check{\mathbf{Q}}^{00}$	$\mathbf{M}\otimes(\check{\mathbf{Q}}^{01}+\check{\mathbf{P}}^{00})$	$\check{\mathbf{V}}$	$\check{\boldsymbol{P}}^{k}=\frac{\mathrm{d}^k\check{\psi}(t_0^+)}{\mathrm{d}t^k}$
TG-UV	$\mathbf{K}\otimes\mathbf{Q}^{01}$	$-\mathbf{K}\otimes\mathbf{Q}^{00}$	$\mathbf{U}$	$\mathbf{Q}^{kl}=\int_I\frac{\mathrm{d}^k\psi}{\mathrm{d}t^k}\otimes\frac{\mathrm{d}^l\psi}{\mathrm{d}t^l}\,\mathrm{dt}$
	$\mathbf{K}\otimes\mathbf{Q}^{00}$	$\mathbf{M}\otimes\mathbf{Q}^{01}$	$\mathbf{V}$	$\boldsymbol{P}^{kl}=\int_I\frac{\mathrm{d}^k\psi}{\mathrm{d}t^k}\frac{\mathrm{d}^l\psi_0}{\mathrm{d}t^l}\,\mathrm{dt}$

TABLE 4.1: Opérateurs du problème espace-temps pour différentes méthodes d'approximation en temps.

- Le problème espace-temps est donné par $[\![\mathbf{A}]\!]\overset{\mathrm{D}}{\cdot}[\mathbf{u}]=[\mathbf{b}]$.
- ϕ est une base éléments finis continus en espace de $\mathscr{U}_h^S(\Omega;0)$ et ϕ_g la fonction de forme associée au degré de liberté imposé non nul.
- $\check{\psi}$ est une base éléments finis discontinus en temps de $\mathscr{U}_{\Delta T}^{\check{T}}(I)$ avec N_T le nombre d'intervalles de temps.
- ψ est une base éléments finis continus en temps de $\mathscr{U}^T(I;0)$ et ψ_0 la fonction de forme associée au noeud situé à $t=0$.

		[**b**]			
Méthode	[**u**]	Neumann	Dirichlet	Dplct. init.	Vitesse init.
Newmark	**U**	$\phi\|_{\partial\Omega_\sigma} \otimes (\mathbf{N}_1.\boldsymbol{p})$	$-\boldsymbol{K}_g \otimes (\mathbf{N}_1.\boldsymbol{g}) - \boldsymbol{M}_g \otimes (\mathbf{N}_1.\ddot{\boldsymbol{g}})$	$(\mathbf{M}.\boldsymbol{U}_0) \otimes \boldsymbol{N}_3 + (\boldsymbol{F}_0 - \mathbf{K}.\boldsymbol{U}_0) \otimes \boldsymbol{N}_5$	$(\mathbf{M}.\boldsymbol{V}_0) \otimes \boldsymbol{N}_4$
TDG-U	$\check{\mathbf{U}}$	$\phi\|_{\partial\Omega_\sigma} \otimes (\check{\mathbf{Q}}^{10}.\check{\boldsymbol{p}})$	$-\boldsymbol{K}_g \otimes (\check{\mathbf{Q}}^{10}.\check{\boldsymbol{g}}) - \boldsymbol{M}_g \otimes (\check{\mathbf{Q}}^{12}.\check{\boldsymbol{g}})$	$(\mathbf{K}.\boldsymbol{U}_0) \otimes \check{\boldsymbol{P}}^0$	$(\mathbf{M}.\boldsymbol{V}_0) \otimes \check{\boldsymbol{P}}^1$
TDG-UV	$\check{\mathbf{U}}$	{}	{}	$(\mathbf{K}.\boldsymbol{U}_0) \otimes \check{\boldsymbol{P}}^0$	{}
	$\check{\mathbf{V}}$	$\phi\|_{\partial\Omega_\sigma} \otimes (\check{\mathbf{Q}}^{00}.\check{\boldsymbol{p}})$	$-\boldsymbol{K}_g \otimes (\check{\mathbf{Q}}^{00}.\check{\boldsymbol{g}}) - \boldsymbol{M}_g \otimes (\check{\mathbf{Q}}^{01}.\dot{\check{\boldsymbol{g}}})$	{}	$(\mathbf{M}.\boldsymbol{V}_0) \otimes \check{\boldsymbol{P}}^0$
TG-UV	**U**	{}	{}	$-(\mathbf{K}.\boldsymbol{U}_0) \otimes \boldsymbol{P}^{01}$	$+(\mathbf{M}.\boldsymbol{V}_0) \otimes \boldsymbol{P}^{00}$
	V	$\phi\|_{\partial\Omega_\sigma} \otimes (\mathbf{Q}^{00}.\boldsymbol{p} + \boldsymbol{P}^{00} p_0)$	$-\boldsymbol{K}_g \otimes (\mathbf{Q}^{00}.\boldsymbol{g}) - \boldsymbol{M}_g \otimes (\mathbf{Q}^{01}.\dot{\boldsymbol{g}})$	$-(\mathbf{K}.\boldsymbol{U}_0) \otimes \boldsymbol{P}^{00}$	$-(\mathbf{M}.\boldsymbol{V}_0) \otimes \boldsymbol{P}^{01}$

TABLE 4.2: (suite)

- Les opérateurs en espace sont donnés par $\mathbf{M} = m(\phi,\phi)$, $\boldsymbol{M}_g = m(\phi,\phi_g)$, $\mathbf{K} = k(\phi,\phi)$ et $\boldsymbol{K}_g = k(\phi,\phi_g)$.
- Les vecteurs $\boldsymbol{p}, \check{\boldsymbol{p}}$ et p_0 sont les coordonnées de $p(t)$ dans $\psi, \check{\psi}$ et ψ_0 (respectivement).
- Les vecteurs $\boldsymbol{g}, \check{\boldsymbol{g}}$ sont les coordonnées de $g(t)$ dans $\psi, \check{\psi}$ (respectivement). Et de façon similaire pour $\dot{\boldsymbol{g}}, \dot{\check{\boldsymbol{g}}}$ et $\ddot{\boldsymbol{g}}$.
- Les vecteurs $\boldsymbol{U}_0, \boldsymbol{V}_0$ sont les coordonnées de $u_0(x)$, $v_0(x)$ (respectivement) dans ϕ.

4.3 Application en élastodynamique

Dans cette section, on illustre la construction des opérateurs $[\![\mathbf{A}]\!]$ et $[\mathbf{b}]$ dans le cas du problème d'élastodynamique bidimensionnel. La principale différence avec le cas unidimensionnel (présenté dans la section précédente) concerne la définition des opérateurs en espace. Aussi, on détaille seulement la construction de ces opérateurs. Le problème espace-temps est alors précisé dans le cas de la méthode de Galerkin discontinue en temps formulée en déplacement-vitesse. On présente ensuite un exemple académique de propagation d'ondes dans un milieu bidimensionnel. La géométrie du cas testé étant très simplifiée, on montre alors comment décomposer les opérateurs en espace, en plus de la décomposition espace-temps.

Remarque 4.9. *Dans le cas du problème d'élastodynamique (bidimensionnel), le champ de déplacement $\boldsymbol{u}(\boldsymbol{x},t)$ est un champ de vecteurs. Il s'exprime en fonction de ses composantes dans un repère $(O,\boldsymbol{e}_1,\boldsymbol{e}_2)$ de $\mathbb{R}^2$ de la façon suivante :*

$$\boldsymbol{u}(\boldsymbol{x},t) = u_1(\boldsymbol{x},t)\boldsymbol{e}_1 + u_2(\boldsymbol{x},t)\boldsymbol{e}_2\,. \tag{4.37}$$

Dans cette section, chaque composante $u_i(\boldsymbol{x},t)$ est traitée de façon indépendante. Et de même pour le champ de vitesse notée $\boldsymbol{v}(\boldsymbol{x},t)$. Aussi, la formulation en déplacement-vitesse est vue comme un problème à quatres champs, à savoir $u_1(\boldsymbol{x},t), u_2(\boldsymbol{x},t), v_1(\boldsymbol{x},t), v_2(\boldsymbol{x},t)$.

4.3.1 Décomposition espace-temps

Dans cette section, on prend $\Omega \subset \mathbb{R}^2$. On note (x_1,x_2) les coordonnées d'un point $\boldsymbol{x} \in \Omega$. Pour simplifier la présentation, on considère seulement le cas où les conditions de Dirichlet et les conditions initiales sont homogènes. On note $\boldsymbol{p}(\boldsymbol{x},t)$ une densité linéïque d'effort[21] appliquée sur la portion $\partial\Omega_\sigma$ de la frontière $\partial\Omega$, et on note $\boldsymbol{n}$ la normale unitaire sortante en tout point de $\partial\Omega$.

Problème de référence

Dans ce cadre, le problème d'élastodynamique consiste à trouver les champs de déplacement $\boldsymbol{u}(\boldsymbol{x},t)(\in\mathbb{R}^2)$ et de contrainte $\boldsymbol{\sigma}(\boldsymbol{x},t)(\in\mathbb{R}^2\otimes\mathbb{R}^2)$ suffisamment réguliés qui vérifient :

- les équations de liaisons et conditions initiales,

$$\boldsymbol{u} = \boldsymbol{0} \qquad \forall(\boldsymbol{x},t)\in\partial\Omega_u\times I\,, \tag{4.38a}$$

$$\boldsymbol{u} = \boldsymbol{0} \qquad \forall(\boldsymbol{x},t)\in\Omega\times\{0\}\,, \tag{4.38b}$$

$$\frac{\partial\boldsymbol{u}}{\partial t} = \boldsymbol{0} \qquad \forall(\boldsymbol{x},t)\in\Omega\times\{0\}\,, \tag{4.38c}$$

21. À ne pas confondre avec le vecteur $\boldsymbol{p}$ de la section précédente.

- les équations d'équilibre,

$$\mathrm{div}(\boldsymbol{\sigma}) = \rho \frac{\partial^2 \boldsymbol{u}}{\partial t^2} \qquad \forall (\boldsymbol{x}, t) \in \Omega \times I\,, \tag{4.38d}$$

$$\boldsymbol{\sigma}.\boldsymbol{n} = \boldsymbol{p} \qquad \forall (\boldsymbol{x}, t) \in \partial\Omega_\sigma \times I\,, \tag{4.38e}$$

- et la relation de comportement,

$$\boldsymbol{\sigma} = \lambda \mathrm{tr}(\boldsymbol{\epsilon})\boldsymbol{I} + 2\mu\boldsymbol{\epsilon} \qquad \forall (\boldsymbol{x}, t) \in \Omega \times I\,, \tag{4.38f}$$

où le champ de déformation $\boldsymbol{\epsilon} \in \mathbb{R}^2 \otimes \mathbb{R}^2$ est donné par la relation suivante :

$$\boldsymbol{\epsilon} = \tfrac{1}{2}(\nabla(\boldsymbol{u}) + \nabla(\boldsymbol{u})')\,. \tag{4.38g}$$

Construction des opérateurs en espace

En suivant la même démarche que dans le Chapitre 1, la formulation faible du problème fait intervenir les produits scalaires $m(,)$, $k(,)$ et $f(;t)$, qui sont définis dans le cas bidimensionnel par

$$m(\boldsymbol{u}^\star, \boldsymbol{u}) = \int_\Omega \rho\, \boldsymbol{u}^\star.\boldsymbol{u}\ \mathrm{d}\Omega\,, \tag{4.39a}$$

$$k(\boldsymbol{u}^\star, \boldsymbol{u}) = \int_\Omega \boldsymbol{\epsilon}(\boldsymbol{u}^\star) : \boldsymbol{\sigma}(\boldsymbol{u})\ \mathrm{d}\Omega\,, \tag{4.39b}$$

$$\text{et} \quad f(\boldsymbol{u}^\star; t) = \int_{\partial\Omega_\sigma} \boldsymbol{u}^\star.\boldsymbol{p}(\boldsymbol{x}, t)\ \mathrm{dS}\,. \tag{4.39c}$$

On se focalise ici sur la construction des opérateurs spatiaux. Chaque composante u_i du vecteur $\boldsymbol{u}$ est exprimée sur la base $\boldsymbol{\phi}$ de l'espace d'approximation $\mathscr{U}_h^S(\Omega;\mathbf{0})$, soit [22]

$$\boldsymbol{u}(\boldsymbol{x}) = (\boldsymbol{U}_1.\boldsymbol{\phi}(\boldsymbol{x}))\boldsymbol{e}_1 + (\boldsymbol{U}_2.\boldsymbol{\phi}(\boldsymbol{x}))\boldsymbol{e}_2\,. \tag{4.40}$$

(et idem pour $\boldsymbol{u}^\star$). On obtient alors les expressions suivantes des matrices de masse et de raideur :

$$\mathbf{M} = \begin{bmatrix} \mathbf{M}_{11} & \mathbf{0} \\ \mathbf{0} & \mathbf{M}_{22} \end{bmatrix} \quad \text{et} \quad \mathbf{K} = \begin{bmatrix} \mathbf{K}_{11} & \mathbf{K}_{12} \\ \text{sym.} & \mathbf{K}_{22} \end{bmatrix} \tag{4.41a}$$

$$\text{avec} \quad \begin{array}{c} \mathbf{K}_{11} = \int_\Omega \left((\lambda+2\mu) \frac{\mathrm{d}\phi}{\mathrm{d}x_1} \otimes \frac{\mathrm{d}\phi}{\mathrm{d}x_1} + \mu \frac{\mathrm{d}\phi}{\mathrm{d}x_2} \otimes \frac{\mathrm{d}\phi}{\mathrm{d}x_2} \right) \mathrm{d}\Omega\,, \\ \mathbf{K}_{22} = \int_\Omega \left((\lambda+2\mu) \frac{\mathrm{d}\phi}{\mathrm{d}x_2} \otimes \frac{\mathrm{d}\phi}{\mathrm{d}x_2} + \mu \frac{\mathrm{d}\phi}{\mathrm{d}x_1} \otimes \frac{\mathrm{d}\phi}{\mathrm{d}x_1} \right) \mathrm{d}\Omega\,, \\ \mathbf{K}_{12} = \int_\Omega \left(\lambda \frac{\mathrm{d}\phi}{\mathrm{d}x_1} \otimes \frac{\mathrm{d}\phi}{\mathrm{d}x_2} + \mu \frac{\mathrm{d}\phi}{\mathrm{d}x_2} \otimes \frac{\mathrm{d}\phi}{\mathrm{d}x_1} \right) \mathrm{d}\Omega\,, \\ \mathbf{M}_{11} = \mathbf{M}_{22} = \int_\Omega \rho\, \phi \otimes \phi\ \mathrm{d}\Omega\,. \end{array} \tag{4.41b}$$

La construction du vecteur des efforts extérieurs est similaire à celle du cas unidimensionnel et n'est donc pas détaillée ici.

22. Avec le choix que l'on a fait pour les conditions de Dirichlet (on impose le déplacement nul dans toutes les directions aux poins $\boldsymbol{x} \in \partial\Omega_u$), on peut prendre le même espace d'approximation pour chaque composante u_i du vecteur déplacement.

Problème espace-temps avec la méthode TDG-UV

En considérant chaque composante des vecteurs déplacement et vitesse comme des champs à part entière, le problème espace-temps formulé avec la méthode TDG-UV s'écrit alors sous la forme du problème à quatres champs suivant [Boucinha *et al.*, 2013a] :

$$\begin{bmatrix} \mathbf{K}_{11}\otimes(\check{\mathbf{Q}}^{01}+\check{\mathbf{P}}^{00}) & \mathbf{K}_{12}\otimes(\check{\mathbf{Q}}^{01}+\check{\mathbf{P}}^{00}) & -\mathbf{K}_{11}\otimes\check{\mathbf{Q}}^{00} & -\mathbf{K}_{12}\otimes\check{\mathbf{Q}}^{00} \\ \mathbf{K}'_{12}\otimes(\check{\mathbf{Q}}^{01}+\check{\mathbf{P}}^{00}) & \mathbf{K}_{22}\otimes(\check{\mathbf{Q}}^{01}+\check{\mathbf{P}}^{00}) & -\mathbf{K}'_{12}\otimes\check{\mathbf{Q}}^{00} & -\mathbf{K}_{22}\otimes\check{\mathbf{Q}}^{00} \\ \mathbf{K}_{11}\otimes\check{\mathbf{Q}}^{00} & \mathbf{K}_{12}\otimes\check{\mathbf{Q}}^{00} & \mathbf{M}_{11}\otimes(\check{\mathbf{Q}}^{01}+\check{\mathbf{P}}^{00}) & \mathbf{0} \\ \mathbf{K}'_{12}\otimes\check{\mathbf{Q}}^{00} & \mathbf{K}_{22}\otimes\check{\mathbf{Q}}^{00} & \mathbf{0} & \mathbf{M}_{22}\otimes(\check{\mathbf{Q}}^{01}+\check{\mathbf{P}}^{00}) \end{bmatrix} \underset{\cdot}{\overset{\mathrm{D}}{\cdot}} \begin{bmatrix} \check{\mathbf{U}}_1 \\ \check{\mathbf{U}}_2 \\ \check{\mathbf{V}}_1 \\ \check{\mathbf{V}}_2 \end{bmatrix} = \begin{bmatrix} \check{\mathbf{0}} \\ \check{\mathbf{0}} \\ \mathbf{I}_1\otimes\check{\mathbf{Q}}^{00}\,\underset{\cdot}{\overset{\mathrm{D}}{\cdot}}\,\check{\mathbf{F}}_1 \\ \mathbf{I}_2\otimes\check{\mathbf{Q}}^{00}\,\underset{\cdot}{\overset{\mathrm{D}}{\cdot}}\,\check{\mathbf{F}}_2 \end{bmatrix} \tag{4.42}$$

où l'on a noté $\check{\mathbf{U}}_1$,$\check{\mathbf{U}}_2$,$\check{\mathbf{V}}_1$ et $\check{\mathbf{V}}_2$ les coordonnées des champs $u_1(\boldsymbol{x},t)$, $u_2(\boldsymbol{x},t)$, $v_1(\boldsymbol{x},t)$ et $v_2(\boldsymbol{x},t)$ dans la base d'approximation $\phi(\boldsymbol{x})\otimes\check{\psi}(t)$ de $\mathscr{U}_h^S(\Omega;0)\otimes\mathscr{U}_{\Delta t}^{\check{T}}(I)$.

Exemple 4.3. (Propagation d'ondes dans un milieu 2D) Dans cet exemple, on décrit un cas test académique de propagation d'ondes dans un milieu bidimensionnel. Ce cas test a été utilisé dans la publication [Boucinha *et al.*, 2013a]. Le domaine spatial est un domaine rectangulaire $\Omega=\Omega_1\times\Omega_2$ avec $\Omega_1=[0,L_1]$ et $\Omega_2=[0,L_2]$. Les conditions aux limites sont choisies de façon à simuler un impact provenant de la gauche de la structure, le bord situé à l'opposé étant encastrée (voir la Figure 4.4). Une densité linéïque d'efforts notée $\boldsymbol{p}(\boldsymbol{x},t)$ est appliquée au centre du bord situé à gauche de la structure (pour $x_1=0$, $x_2\in[\frac{L_2}{4},\frac{3L_2}{4}]$). Cette effort est donné sous la forme $\boldsymbol{p}(\boldsymbol{x},t)=p^S(\boldsymbol{x})p^T(t)\boldsymbol{e}_1$. La composante spatiale $p^S(\boldsymbol{x})$ est fixée constante, et la composante temporelle $p^T(t)$ est prise sous la forme d'un choc d'une durée ΔT, c'est-à-dire :

$$p^T(t)=\begin{cases}\frac{1}{2}(1-\cos(\frac{2\pi}{\Delta T}t)) & \text{si} \quad t\in[0,\Delta T],\\ 0 \quad \text{sinon}.\end{cases} \tag{4.43}$$

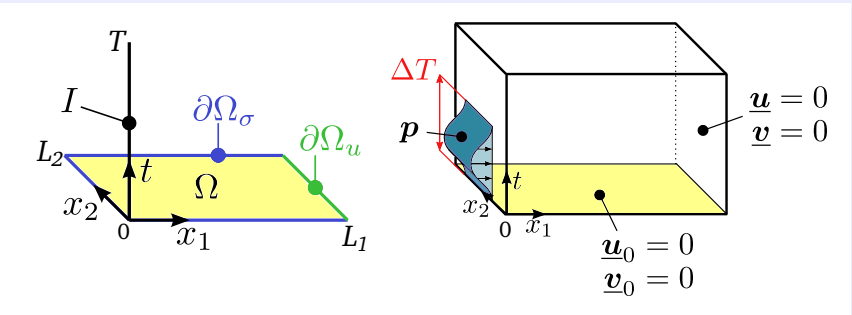

FIGURE 4.4: Description du cas test bidimensionnel.

Les paramètres géométriques sont fixés à $L_1=1$m et $L_2=0.5$m. Le module d'élasticité est pris à $E=200$GPa et le coefficient de poisson à $\nu=0.3$, la masse volumique $\rho=8000\text{kg/m}^3$. Avec ces paramètres, les ondes longitudinales se propage à la vitesse $c_L\simeq 5800$m/s et les ondes transversales à la vitesse $c_T\simeq 3100$m/s. La simulation est réalisée

sur une durée $T = 1$ms. De cette façon, une onde longitudinale parcours environ 6 fois la longueur L_1.

Afin de comparer différents régimes dynamiques, on fait varier la durée ΔT du choc. En suivant la démarche proposée au Chapitre 1, on introduit le nombre sans dimension, noté κ tel que

$$\kappa = \Big(\frac{L_1}{c_L \Delta T}\Big)^2, \tag{4.44}$$

qui caractérise le régime dynamique transitoire, associé à la propagation d'une onde longitudinale suivant $\boldsymbol{e}_1$. Les valeurs de κ obtenues avec les valeurs de ΔT prises dans [Boucinha *et al.*, 2013a] sont données dans le Tableau 4.3.

ΔT(ms)	κ	$N_1 \times N_2 \times N_T$	err^{disc}(%)	
			u	v
1	0.03	$8 \times 4 \times 40$	< 2	< 2
0.2	0.7	$32 \times 16 \times 160$	< 2	< 5
0.05	12	$128 \times 64 \times 640$	< 2	< 9
0.01	298	$320 \times 160 \times 1600$	< 4	< 25

TABLE 4.3: Paramères du cas test bidimensionnel.

Le problème est résolu avec des éléments finis P1 en espace (éléments rectangulaires) et le schéma TDG P1-P1 en temps. Aussi, pour chaque valeur de κ, on fixe les paramètres du maillage espace-temps de façon à ce que l'erreur de discrétisation (sur le champ de déplacement) soit inférieure à 4%. Cette erreur est évaluée, sur tout le domaine espace-temps, de la façon suivante :

$$\text{err}^{disc} = \frac{\sqrt{\| u_1 - u_1^{h,\Delta t} \|_2^2 + \| u_2 - u_2^{h,\Delta t} \|_2^2}}{\sqrt{\| u_1 \|_2^2 + \| u_2 \|_2^2}}, \tag{4.45}$$

où u_1 et u_2 sont les composantes d'une solution de référence $\boldsymbol{u}$ calculée sur un maillage très fin, et $\| \ \|_2$ est la norme canonique définie au Chapitre 2. Le nombre d'éléments du maillage spatial (N_1 dans la direction $\boldsymbol{e}_1$ et N_2 dans la direction $\boldsymbol{e}_2$) ainsi que le nombre d'intervalle de temps (N_T) sont donnés dans le Tableau 4.3 pour les différentes valeurs de κ. L'erreur de discrétisation sur le champ de vitesse obtenue avec ces paramètres du maillage est également précisée dans le Tableau 4.3. On remarquera que l'erreur sur le champ de vitesse est plus importante que l'erreur sur le champ de déplacement. Il serait donc intéressant d'utiliser des espaces d'approximation différents pour chaque champ. On se contente ici de prendre le même maillage espace-temps pour ces deux champs.

Les résultats obtenus pour les différentes valeurs de κ sont présentés sur la Figure 4.5 et la Figure 4.6. Sur la Figure 4.5, les amplitudes du champ de déplacement et de

vitesse sont représentées sur le domaine espace-temps. Plus κ augmente et plus on observe de forts gradients par rapport aux variables spatiales et temporelle.

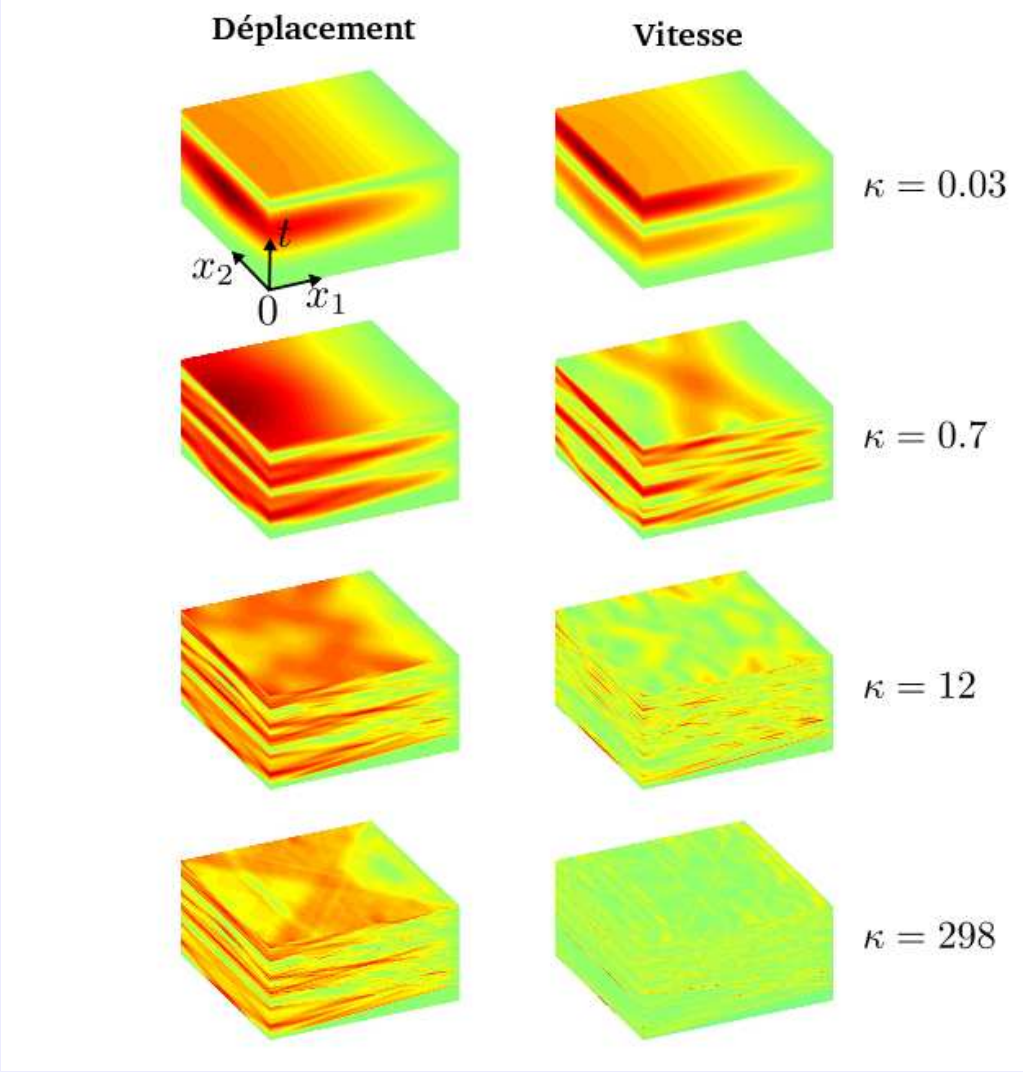

FIGURE 4.5: Amplitudes des champs de déplacement et de vitesse dans le domaine espace-temps.

La Figure 4.6 montre les isovaleurs de l'amplitude du champ de déplacement à différents instants, pour les valeurs de $\kappa = 0.7 - 12 - 298$. L'amplitude vaut zéro pour l'isovaleur verte et un pour l'isovaleur rouge foncée. Les instants $t = 2.5$ à $t = 20 * 1e - 5$s correspondent au premier trajet d'une onde longitudinale dans la direction e_1. Globalement, on observe que l'augmentation de la valeur de κ fait apparaître des motifs dont la forme est de plus en plus complexes, et peut évoluer très soudainement au cours du temps (voir les instants $t > 20$).

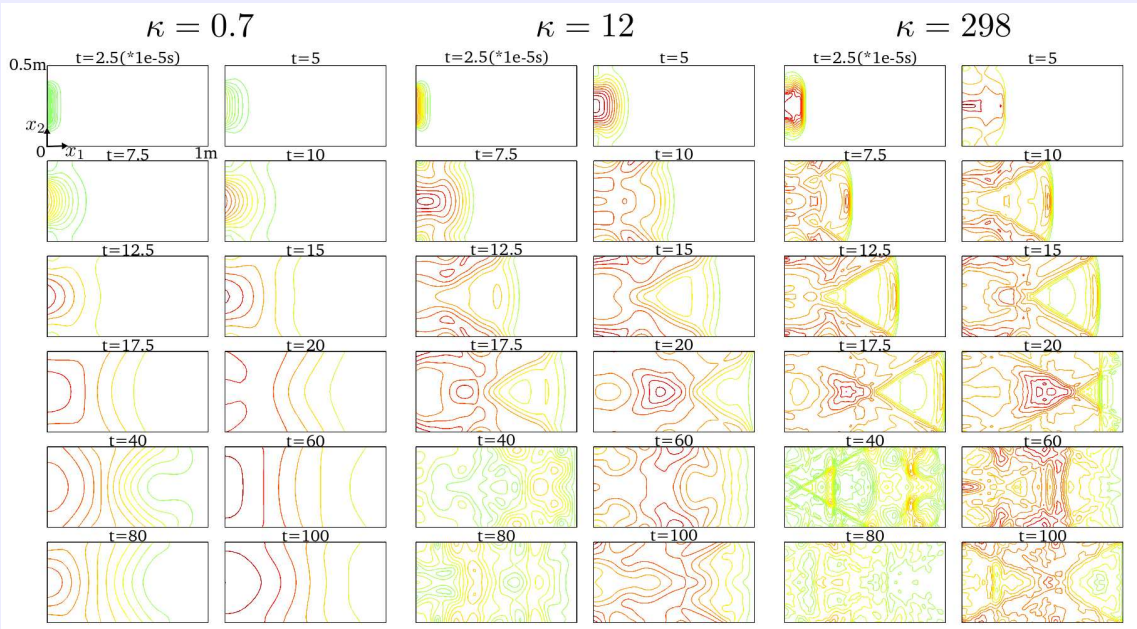

FIGURE 4.6: Isovaleurs de l'amplitude du champ de déplacement, représentées à différents instants dans le domaine spatial $\Omega_1 \times \Omega_2$, pour $\kappa = 0.7 - 12 - 298$.

4.3.2 Décomposition espace-espace-temps

On montre ici comment construire une représentation séparée des opérateurs spatiaux. Une telle représentation peut notamment être utilisée pour construire des éléments finis structuraux à comportement volumique [Bognet *et al.*, 2012]. Elle est simplement utilisée ici pour calculer les matrices de masse et de raideur de façon analytique. La construction est détaillée en se focalisant sur les variables spatiales (on omet le temps).

L'idée est d'approcher chaque composante $u_i(x_1, x_2)$ du champ de déplacement sous la forme

$$u_i(x_1, x_2) = \phi^1(x_1) \otimes \phi^2(x_2) \overset{\mathrm{D}}{\cdot} \mathbf{u}_i \quad \text{avec} \quad \mathbf{u}_i \in \mathbb{R}^{n_S(1) \times n_S(2)}, \tag{4.46}$$

où $\phi^1(x_1)$ est une base de l'espace [23] d'approximation $\mathscr{U}^{S_1}(\Omega_1)$, et $\phi^2(x_2)$ est une base de l'espace d'approximation $\mathscr{U}^{S_2}(\Omega_2)$. En d'autres termes, l'espace $\mathscr{U}^{S}(\Omega_1 \times \Omega_2)$ est construit par tensorisation des espaces $\mathscr{U}^{S_1}(\Omega_1)$ et $\mathscr{U}^{S_2}(\Omega_2)$.

En remplaçant cette expression du champ de déplacement (et en prenant la même représentation pour le champ test) dans les produits scalaires $m(,)$ et $k(,)$, on peut

23. On omet ici l'indice h utilisé dans le manuscrit pour spécifier un espace de dimension finie.

alors montrer que les composantes des matrices de masse et de raideur s'écrivent sous la forme séparée suivante :

$$\mathbf{K}_{11} = (\lambda + 2\mu)\,\mathbf{S}^{1(11)} \otimes \mathbf{S}^{2(00)} + \mu\,\mathbf{S}^{1(00)} \otimes \mathbf{S}^{2(11)}\,, \tag{4.47a}$$

$$\mathbf{K}_{22} = (\lambda + 2\mu)\,\mathbf{S}^{1(00)} \otimes \mathbf{S}^{2(11)} + \mu\,\mathbf{S}^{1(11)} \otimes \mathbf{S}^{2(00)}\,, \tag{4.47b}$$

$$\mathbf{K}_{12} = \lambda\,\mathbf{S}^{1(10)} \otimes \mathbf{S}^{2(01)} + \mu\,\mathbf{S}^{1(01)} \otimes \mathbf{S}^{2(10)}\,, \tag{4.47c}$$

$$\mathbf{M}_{11} = \mathbf{M}_{22} = \rho\,\mathbf{S}^{1(00)} \otimes \mathbf{S}^{2(00)}\,, \tag{4.47d}$$

avec l'expression suivante de la matrice $\mathbf{S}^{d(pq)}$

$$\mathbf{S}^{d(pq)} = \int_{\Omega_d} \frac{\mathrm{d}^p(\phi^d)}{\mathrm{d}x_d^p} \otimes \frac{\mathrm{d}^q(\phi^d)}{\mathrm{d}x_d^q}\,\mathrm{dx_d}\,.$$

Remarque 4.10. *Ce type de représentation peut être appliqué seulement sur des géométries particulières. Grossièrement, on doit pouvoir séparer une intégrale définie sur* Ω *en un produit de deux intégrales définies sur* Ω_1 *et* Ω_2*, respectivement*[24]*. On pourra consulter les travaux de [Bognet* et al.*, 2012] où les auteurs introduisent une fonction indicatrice permettant de prendre en compte des géométries plus compliquées.*

Remarque 4.11. *La représentation du champ de déplacement sur la forme de l'équation (4.46) rend plus difficile la prise en compte de conditions de Dirichlet dans l'espace d'approximation. Notamment, on ne peut pas appliquer une condition de Dirichlet localisé sur un bord du domaine* Ω*. Des stratégies permettant de s'affranchir de cette contrainte ont notamment été proposées par [González* et al.*, 2010]. Une autre solution est d'imposer les conditions de Dirichlet de façon faible, comme proposée par [Ammar* et al.*, 2012].*

Remarque 4.12. *La condition de Dirichlet utilisée dans l'Exemple 4.3 peut être imposée en prenant* $\mathscr{U}^S(\Omega;0) = \mathscr{U}^{S_1}(\Omega_1;0) \times \mathscr{U}^{S_2}(\Omega_2)$*.*

Remarque 4.13. *En pratique, la matrice* $\mathbf{S}^{d(pq)}$ *est calculée analytiquement, puis on utilise le produit de Kronecker pour assembler les matrices de masse et de raideur sous la forme de (4.41b).*

4.4 Conclusion

Dans ce chapitre, un formalisme générique, basé sur une représentation tensoriel du problème espace-temps, a été introduit dans un cadre multichamps. La construction des opérateurs du problème espace-temps a été illustrée dans le cas d'une discrétisation de l'équation des ondes et du problème d'élastodynamique, à l'aide des

24. Si une telle séparation est possible par l'intermédiaire d'un changement de variables, on doit pouvoir également écrire le jacobien sous la forme d'un produit de fonctions définies sur Ω_1 et Ω_2 respectivement.

méthodes classiques d'approximation introduites dans le Chapitre 1 (méthode éléments finis en espace et en temps, schéma d'intégration de Newmark). Une attention particulière a été apportée au traitement des conditions aux limites (conditions de Dirichlet et Neumann, conditions initiales en déplacement et vitesse). Un exemple académique de propagation d'ondes dans un milieu bidimentionnel a ensuite été présenté.

De cette façon, le problème de dynamique discrétisé en espace et en temps s'écrit sous la forme d'un unique système linéaire. Bien sûr, la résolution de ce système linéaire à l'aide des solveurs classiques n'est pas envisageable puisque le système linéaire à résoudre est de taille $n_S n_T \times n_S n_T$, où n_S et n_T sont les dimensions des espaces d'approximation spatiale et temporelle respectivement. Aussi, dans le chapitre suivant, on montre comment construire une approximation de la solution de ce système linéaire sous la forme d'une représentation à variables séparées espace-temps. Cette construction est basée sur la décomposition généralisée propre d'un tenseur, mieux connue sous son acronyme anglais PGD « Proper Generalized Decomposition ».

Les paramètres d'entrée du solveur proposé sont les opérateurs du problème espace-temps donnés sous la forme de l'équation (4.4). Ce solveur peut alors être utilisé comme une « boîte noire » pour obtenir une approximation de rang M, dans une complexité de l'ordre de $M\xi\big(\mathbf{lin}(n_S)+\mathbf{lin}(n_T)\big)$ opérations, où ξ est un nombre d'itérations et $\mathbf{lin}(n)$ est la complexité de la résolution d'un système linéaire de taille $n \times n$. Ce solveur est clairement compétitif par rapport à une résolution brutale du problème espace-temps.

L'objectif du prochain chapitre est d'évaluer l'efficacité d'un tel solveur dans le cas d'un problème de choc. On ouvre la boîte noire et on décortique tous les composants.

Chapitre 5

État de l'art sur la décomposition généralisée propre

Dans ce chapitre, on établit un état de l'art des algorithmes utilisés, dans le cadre de la méthode de décomposition généralisée propre (PGD), pour construire une approximation à variables séparées espace-temps, de la solution d'un système linéaire lui même donné sous format tensoriel.

Sommaire

Dans le chapitre précédent, on a montré comment écrire le problème de dynamique transitoire, discrétisé en espace et en temps, sous la forme d'un unique système linéaire, dont les opérateurs sont donnés sous format tensoriel. On montre ici comment approcher la solution de ce problème, sous la forme d'une représentation à variables séparées espace-temps, sans connaissance a priori sur la solution du problème. La construction de cette approximation est basée sur la décomposition généralisée propre (PGD).

5.1 Introduction

La PGD a été initiée dans le cadre de la méthode à grand incrément de temps (méthode LATIN) pour résoudre efficacement des problèmes de mécaniques non linéaires [Ladevèze, 1999]. Dans ce cadre, elle permet de diminuer les coûts numériques associés à des résolutions répétées d'un problème linéaire global défini sur le domaine espace-temps. Dans les premiers travaux [Dureisseix *et al.*, 2003, Ladevèze et Nouy, 2003], la construction de l'approximation à variables séparées espace-temps est associée à un critère de Galerkin. Ce critère permet de séparer le problème espace-temps (linéaire) en deux problèmes (non-linéaires) définis l'un en espace et l'autre en temps. On remplace de cette façon la résolution d'un problème de taille $n_S n_T \times n_S n_T$ par plusieurs résolutions alternatives de deux problèmes de tailles $n_S \times n_S$ et $n_T \times n_T$ respectivement. L'efficacité de la méthode dépend alors du nombre de fois que l'on résoud ces problèmes de plus petites tailles. En pratique, la convergence (partielle) est obtenue après quelques itérations (deux ou trois), et on passe au calcul du mode espace-temps suivant. Pour les problèmes non-symétriques, la robustesse de la méthode est améliorée en introduisant la notion de meilleure approximation au sens de la minimisation de l'erreur en relation de comportement [Nouy et Ladevèze, 2004, Ladevèze *et al.*, 2010, Passieux *et al.*, 2010]. Dans le cadre de la méthode LATIN, l'efficacité de la PGD tient également au fait que l'on ait à résoudre plusieurs fois, un système linéaire global dont la solution varie peu entre les différentes résolutions. Aussi, on peut tirer parti de l'approximation de la solution du système linéaire précédent, pour initialiser la nouvelle approximation à variables séparées du problème courant.

Depuis une dizaine d'années, le potentiel de la PGD a été révélé dans de nombreux domaines [Chinesta *et al.*, 2011]. Une approximation à variables séparées plus générale a notamment été considérée par [Ammar *et al.*, 2006] sous la forme,

$$u(x_1, \cdots, x_D) \simeq u_M(x_1, \cdots, x_D) = \sum_{m=1}^{M} \prod_{d=1}^{D} w_m^d(x_d)\,, \tag{5.1}$$

pour approcher la solution de problèmes multidimensionnels. Pour ce type de problèmes, les méthodes classiques d'approximation se heurtent rapidement à la « malédiction de la dimensionnalité » et l'approximation à variables séparées s'avère des plus

adaptée [Chinesta *et al.*, 2008]. On pourra consulter [Chinesta *et al.*, 2010] pour une revue des techniques récemment développées, ainsi que [Nouy, 2010b] pour la formalisation de la PGD dans le cadre des problèmes stochastiques en dimensions élévées.

Très peu de travaux existent, concernant des applications de la PGD à des problèmes de dynamique des structures ou de propagation d'ondes. On citera : l'approche de [Chevreuil et Nouy, 2012] pour la quantification d'incertitudes en dynamique des structures [1], le couplage entre la PGD et la théorie variationnelle des rayons complexes (TVRC) [2] proposé par [Barbarulo, 2012] dans le cadre de problème d'acoustique moyenne fréquence, l'application à l'équation d'Helmholtz [3] par [Modesto *et al.*, 2012] dans le cadre d'études paramétriques de problèmes réalistes de propagation d'ondes. L'approximation à variables séparées espace-temps a été appliquée à la résolution de problèmes d'évolution de type parabolique par [Ammar *et al.*, 2007, Nouy, 2010a, Bonithon *et al.*, 2011]. Elle a été appliquée, dans le cadre des présents travaux, à la résolution de l'équation des ondes [Boucinha *et al.*, 2013b] et du problème d'élastodynamique 2D [Boucinha *et al.*, 2013a], formulés sur le domaine espace-temps.

Remarque 5.1. *Un des points clés des stratégies basées sur la PGD, concerne le choix du rang M de l'approximation. Dans ce manuscrit, on suppose que le rang M est connu a priori, et fixé une fois pour toute. Ce choix est très restrictif si l'on considère des applications réelles. Dans ce cas, il s'avère indispensable d'introduire des estimateurs d'erreur permettant de comparer l'erreur due à l'approximation de rang M, à celle due à la discrétisation. La mise en oeuvre des méthodes classiques de construction d'un estimateur d'erreur dans le cadre de la PGD a notamment été considérée par [Ammar* et al.*, 2010a, Ladevèze et Chamoin, 2011, Almeida, 2013, Billaud-Friess* et al.*, 2013].*

5.2 Définitions de la PGD

Différentes définitions de la PGD ont été proposées dans la littérature. Les définitions classiques, basées sur un critère de Galerkin [Ladevèze, 1999, Ammar *et al.*, 2006, Ammar *et al.*, 2007, Nouy, 2007, Nouy, 2008] ou bien sur la minimisation du résidu [Nouy et Ladevèze, 2004, Beylkin et Mohlenkamp, 2005, Ammar, 2010], ainsi qu'une nouvelle définition basée sur un critère de Petrov-Galerkin ont été formalisées par [Nouy, 2010a] dans le cadre des problèmes d'évolution. Pour chacune de ces défini-

1. Le problème est formulé en fréquence, et pour une fréquence donné, le champ de déplacement (complexe) est approché sous la forme $u_M(x,p) = \sum_{m=1}^{M} w_m(x)\lambda_m(p)$ où x représente l'espace physique et p un paramètre aléatoire modélisant les incertitudes.

2. Le vecteur des inconnues $\boldsymbol{U}(\omega)$ associé à la formulation TVRC (qui sont les amplitudes des rayons de vibration) est approché sous la forme $\boldsymbol{U}_M(\omega) = \sum_{m=1}^{M} \boldsymbol{X}_m\lambda_m(\omega)$, où ω est la fréquence associée à un problème de vibrations forcées.

3. L'amplitude du champ de déplacement est approchée sous la forme $u_M(x,k,\theta) = \sum_{m=1}^{M} w_m^x(x)\, w_m^k(k)\, w_m^\theta(\theta)$ où x représente l'espace physique, k est le nombre d'onde et θ un paramètre lié aux conditions aux limites.

tions, de nombreux algorithmes ont été proposés. Les algorithmes utilisés dans la suite du manuscrit sont présentés dans cette section.

Dans cette présentation, on se place du point de vue discret. On suppose que le problème de référence a été discrétisé en espace et en temps, et que les opérateurs du système linéaire espace-temps sont donnés sous format tensoriel. On rappelle que le problème espace-temps consiste à trouver $\mathbf{u} \in \mathbb{R}^{n_S} \otimes \mathbb{R}^{n_T}$ tel que

$$\mathbf{A} \overset{\mathrm{D}}{\cdot} \mathbf{u} = \mathbf{b}, \tag{5.2}$$

où les opérateurs [4] $\mathbf{A} \in \mathscr{L}(\mathbb{R}^{n_S} \otimes \mathbb{R}^{n_T})$ et $\mathbf{b} \in \mathbb{R}^{n_S} \otimes \mathbb{R}^{n_T}$ sont donnés sous la forme

$$\mathbf{A} = \sum_{k=1}^{M_A} \mathbf{A}_k^S \otimes \mathbf{A}_k^T \quad \text{et} \quad \mathbf{b} = \sum_{k=1}^{M_b} \boldsymbol{b}_k^S \otimes \boldsymbol{b}_k^T. \tag{5.3}$$

L'objectif de la PGD est alors de trouver une approximation $\mathbf{u}_M$ de $\mathbf{u}$, sans autres connaissances a priori sur le tenseur $\mathbf{u}$ que les opérateurs du problème dont il est solution. Cette approximation est cherchée, dans le sous-ensemble, noté R_M, des décompositions espace-temps de rang M, défini par

$$\mathrm{R}_M = \left\{ \mathbf{u} \in \mathbb{R}^{n_S} \otimes \mathbb{R}^{n_T} \mid \mathbf{u} = \sum_{m=1}^{M} \boldsymbol{w}_m \otimes \boldsymbol{\lambda}_m \text{ avec } \boldsymbol{w}_m \in \mathbb{R}^{n_S}, \boldsymbol{\lambda}_m \in \mathbb{R}^{n_T} \right\}. \tag{5.4}$$

Afin de simplifier la présentation, les différentes définitions de la PGD vont être introduites dans le cas d'une approximation de rang un. La construction d'une approximation dans R_M sera ensuite précisée.

Enfin, on rappelle que la meilleure approximation $\mathbf{u}_M$ de $\mathbf{u}$ dans R_M a été définie, au Chapitre 2, comme la solution d'un problème de minimisation par rapport à la norme $\| \, \|_2$ (associée au produit scalaire canonique [5]), c'est-à-dire,

$$\boxed{\mathbf{u}_M = \arg \min_{\mathbf{u}^\star \in \mathrm{R}_M} \| \mathbf{u} - \mathbf{u}^\star \|_2}. \tag{5.5}$$

La PGD va également être définie au sens d'un problème de meilleure approximation, mais par rapport à une norme plus générale (qui ne vérifie pas nécessairement la propriété de séparabilité (2.10)). On comparera à la fin de ce chapitre, l'approximation obtenue avec la PGD et la meilleure approximation de rang M au sens de la norme $\| \, \|_2$, qui est la meilleure approximation de référence.

4. On note $\mathscr{L}(\mathbb{R}^{n_S} \otimes \mathbb{R}^{n_T})$ l'espace des opérateurs linéaires sur $\mathbb{R}^{n_S} \otimes \mathbb{R}^{n_T}$.
5. On rappelle que $\langle \mathbf{u}, \mathbf{v} \rangle_2 = \mathbf{u} \overset{\mathrm{D}}{\cdot} \mathbf{v} = \sum_{i=1}^{n_S} \sum_{j=1}^{n_T} \mathrm{u}_{ij} \mathrm{v}_{ij}$, $\forall \mathbf{u}, \mathbf{v} \in \mathbb{R}^{n_S} \otimes \mathbb{R}^{n_T}$.

5.2.1 Critère d'orthogonalité de Galerkin

La façon la plus naturelle de définir la PGD du tenseur $\mathbf{u}$ est d'imposer l'annulation du résidu au sens d'un critère d'orthogonalité de Galerkin. On note cette définition (G)PGD. L'approximation de rang un est alors définie comme le mode espace-temps $w \otimes \lambda$ qui vérifie le critère d'orthogonalité suivant $\forall w^\star \in \mathbb{R}^{n_S}$, $\forall \lambda^\star \in \mathbb{R}^{n_T}$,

$$\boxed{\langle w^\star \otimes \lambda + w \otimes \lambda^\star , \mathbf{A} \overset{D}{\cdot} w \otimes \lambda - \mathbf{b} \rangle_2 = 0}. \tag{5.6}$$

Lorsque l'opérateur $\mathbf{A}$ est symétrique définie positif, on peut lui associer le produit scalaire $\langle , \rangle_\mathbf{A}$ définie sur $\mathbb{R}^{n_S} \otimes \mathbb{R}^{n_T}$ par $\langle \mathbf{u}, \mathbf{v} \rangle_\mathbf{A} = \langle \mathbf{u}, \mathbf{A} \overset{D}{\cdot} \mathbf{v} \rangle_2 \; \forall \mathbf{u}, \mathbf{v}$. On note $\| \; \|_\mathbf{A}$ la norme associée. Dans ce cas, le mode espace-temps $w \otimes \lambda$ définie par le critère d'orthogonalité (5.6) est également la meilleure approximation de $\mathbf{u}$ dans R_1 au sens du problème de minimisation suivant :

$$w \otimes \lambda = \arg \min_{\mathbf{u}^\star \in \mathrm{R}_1} \| \mathbf{u} - \mathbf{u}^\star \|_\mathbf{A}. \tag{5.7}$$

Remarque 5.2. *La PGD associée au critère d'orthogonalité de Galerkin est définie a priori par l'équation (5.6) et a posteriori par l'équation (5.7). On peut montrer l'équivalence entre les deux définitions en introduisant la fonctionnelle* $J : \mathbf{u}^\star \rightarrow \frac{1}{2} \| \mathbf{u} - \mathbf{u}^\star \|^2_\mathbf{A}$ *associée au problème de minimisation (5.7). La condition de stationnarité de* J *en* $w \otimes \lambda$ *est donnée par :*

$$\begin{aligned}
& \frac{\delta J(w \otimes \lambda)}{\delta (w \otimes \lambda)} \overset{D}{\cdot} \delta (w \otimes \lambda) = 0, && \forall \delta (w \otimes \lambda), \\
\Leftrightarrow \quad & \frac{\delta J(w \otimes \lambda)}{\delta w} . \delta w + \frac{\delta J(w \otimes \lambda)}{\delta \lambda} . \delta \lambda = 0, && \forall \delta w, \forall \delta \lambda, \\
\Leftrightarrow \quad & \langle \delta w \otimes \lambda + w \otimes \delta \lambda , w \otimes \lambda - \mathbf{u} \rangle_\mathbf{A} = 0, && \forall \delta w, \forall \delta \lambda, \\
\Leftrightarrow \quad & \langle \delta w \otimes \lambda + w \otimes \delta \lambda , \mathbf{A} \overset{D}{\cdot} w \otimes \lambda - \mathbf{b} \rangle_2 = 0, && \forall \delta w, \forall \delta \lambda.
\end{aligned}$$

Le mode espace-temps, solution de (5.6), vérifie donc également la condition de stationnarité associée au problème de minimisation (5.7).

Remarque 5.3. *La définition (5.6) est également équivalente au problème de minimisation suivant :*

$$w \otimes \lambda = arg \min_{\mathbf{u}^\star \in \mathrm{R}_1} \tfrac{1}{2} \langle \mathbf{u}^\star , \mathbf{u}^\star \rangle_\mathbf{A} - \langle \mathbf{u}^\star , \mathbf{b} \rangle_2 . \tag{5.8}$$

5.2.2 Minimisation du résidu

Lorsque l'opérateur $\mathbf{A}$ est non-symétrique, une définition plus robuste de la PGD est obtenue en minimisant le résidu dans une certaine norme. Classiquement, on choisit la norme $\| \; \|_2$. On notera (R)PGD cette définition. L'approximation de rang un est alors définie comme le mode espace-temps $w \otimes \lambda$ qui vérifie

$$\boxed{w \otimes \lambda = \arg \min_{\mathbf{u}^\star \in \mathrm{R}_1} \| \mathbf{b} - \mathbf{A} \overset{D}{\cdot} \mathbf{u}^\star \|_2}. \tag{5.9}$$

Cette définition est donnée a priori. Une définition a posteriori équivalente peut être obtenue en remplaçant $\mathbf{b}$ par $\mathbf{A}\dot{\mathrm{D}}\mathbf{u}$ dans (5.9). En notant $\mathbf{A}'$ l'opérateur transposée de $\mathbf{A}$ et en introduisant la norme $\|\ \|_{\mathbf{A}'\dot{\mathrm{D}}\mathbf{A}}$ associée au produit scalaire $\langle\,,\,\rangle_{\mathbf{A}'\dot{\mathrm{D}}\mathbf{A}}$ défini par $\langle\mathbf{u},\mathbf{v}\rangle_{\mathbf{A}'\dot{\mathrm{D}}\mathbf{A}}=\langle\mathbf{u},\mathbf{A}'\dot{\mathrm{D}}\mathbf{A}\dot{\mathrm{D}}\mathbf{v}\rangle_2\ \forall\mathbf{u},\mathbf{v}$, on obtient $\|\mathbf{b}-\mathbf{A}\dot{\mathrm{D}}\mathbf{u}^\star\|_2=\|\mathbf{u}-\mathbf{u}^\star\|_{\mathbf{A}'\dot{\mathrm{D}}\mathbf{A}}$. Aussi, la définition a posteriori de la PGD en minimum de résidu est donnée par :

$$\boldsymbol{w}\otimes\boldsymbol{\lambda}=\arg\min_{\mathbf{u}^\star\in\mathrm{R}_1}\|\mathbf{u}-\mathbf{u}^\star\|_{\mathbf{A}'\dot{\mathrm{D}}\mathbf{A}}. \tag{5.10}$$

5.2.3 Critère de Petrov-Galerkin

Les approximations obtenues avec les définitions précédentes peuvent être assez éloignées de la meilleure approximation définie dans la norme $\|\ \|_2$. Aussi, une nouvelle définition a été introduite par [Nouy, 2010a] afin d'améliorer la précision de l'approximation PGD, par rapport à la meilleure approximation dans une norme de référence (qui est ici la norme $\|\ \|_2$). Cette nouvelle définition est basée sur un critère de Petrov-Galerkin (on l'a notera (PG)PGD). L'idée est d'imposer l'orthogonalité du résidu par rapport à un autre ensemble de modes espace-temps (pris également de rang un), que l'on obtient en résolvant un problème adjoint. On cherche alors le mode $\boldsymbol{w}\otimes\boldsymbol{\lambda}$, ainsi qu'un autre mode $\tilde{\boldsymbol{w}}\otimes\tilde{\boldsymbol{\lambda}}$ qui vérifient les critères d'orthogonalité suivants $\forall\boldsymbol{w}^\star\in\mathbb{R}^{n_S}$, $\forall\boldsymbol{\lambda}^\star\in\mathbb{R}^{n_T}$, $\forall\tilde{\boldsymbol{w}}^\star\in\mathbb{R}^{n_S}$, $\forall\tilde{\boldsymbol{\lambda}}^\star\in\mathbb{R}^{n_T}$,

$$\langle\tilde{\boldsymbol{w}}^\star\otimes\tilde{\boldsymbol{\lambda}}+\tilde{\boldsymbol{w}}\otimes\tilde{\boldsymbol{\lambda}}^\star,\mathbf{A}\dot{\mathrm{D}}\boldsymbol{w}\otimes\boldsymbol{\lambda}-\mathbf{b}\rangle_2=0, \tag{5.11a}$$

$$\langle\mathbf{A}'\dot{\mathrm{D}}\tilde{\boldsymbol{w}}\otimes\tilde{\boldsymbol{\lambda}}-\boldsymbol{w}\otimes\boldsymbol{\lambda},\boldsymbol{w}^\star\otimes\boldsymbol{\lambda}+\boldsymbol{w}\otimes\boldsymbol{\lambda}^\star\rangle_2=0. \tag{5.11b}$$

Remarque 5.4. *Le critère de Petrov-Galerkin peut également être appliqué en considérant que le problème espace-temps est le problème symétrisé défini par*

$$\mathbf{A}'\dot{D}\mathbf{A}\dot{D}\mathbf{u}=\mathbf{A}\dot{D}\mathbf{b}. \tag{5.12}$$

On appelera cette définition (PGsym)PGD.

5.2.4 Bilan

Dans un cadre général, les différentes définitions de la PGD peuvent s'écrire sous la forme d'un problème de minimisation. La meilleure approximation de rang M de $\mathbf{u}$ est cherchée dans R_M, comme solution du problème de minimisation suivant :

$$\boxed{\mathbf{u}_M=\arg\min_{\mathbf{u}^\star\in\mathrm{R}_M}J(\mathbf{u}^\star)}, \tag{5.13}$$

où la fonctionnelle J est donnée, sous forme a priori ou a posteriori, dans le Tableau 5.1 pour les définitions classiques [6] de la PGD. On peut comparer les définitions a posteriori de la PGD, avec les décompositions classiques (SVD et POD). Alors que la SVD et la

6. Le critère de Petrov-Galerkin est associé à un problème de « min-max » (voir [Nouy, 2010b]) et n'est pas inséré dans le tableau.

	a priori $J(\mathbf{u}^\star;\mathbf{A},\mathbf{b})$	**a posteriori** $J(\mathbf{u}^\star;\mathbf{u})$
(G)PGD	$\frac{1}{2}\langle \mathbf{u}^\star , \mathbf{u}^\star \rangle_{\mathbf{A}} - \langle \mathbf{u}^\star , \mathbf{b} \rangle_2$	$\frac{1}{2}\left\| \mathbf{u}-\mathbf{u}^\star \right\|^2_{\mathbf{A}}$
(R)PGD	$\frac{1}{2}\left\| \mathbf{b}-\mathbf{A}\cdot\mathbf{u}^\star \right\|^2_2$	$\frac{1}{2}\left\| \mathbf{u}-\mathbf{u}^\star \right\|^2_{\mathbf{A}'\cdot\mathbf{A}}$
SVD	$\frac{1}{2}\left\| \mathbf{b}-\mathbf{A}\cdot\mathbf{u}^\star \right\|^2_{(\mathbf{A}\cdot\mathbf{A}')^{-1}}$	$\frac{1}{2}\left\| \mathbf{u}-\mathbf{u}^\star \right\|^2_2$
POD	$\frac{1}{2}\left\| \mathbf{b}-\mathbf{A}\cdot\mathbf{u}^\star \right\|^2_{(\mathbf{A}\cdot\mathbf{N}^{-1}\cdot\mathbf{A}')^{-1}}$	$\frac{1}{2}\left\| \mathbf{u}-\mathbf{u}^\star \right\|^2_{\mathbf{N}}$

TABLE 5.1: Définitions a priori et a posteriori de la fonctionnelle J associée à différentes décompositions.

POD définissent la meilleure approximation au sens d'une norme vérifiant la propriété de séparabilité (2.10) (l'opérateur $\mathbf{N}$ est tel que $\mathbf{N}=\mathbf{N}^S\otimes\mathbf{N}^T$), la PGD définit la meilleure approximation au sens d'une norme plus générale associée à l'opérateur du problème (dans le cas général, $\mathbf{A}\neq\mathbf{A}^S\otimes\mathbf{A}^T$). D'autre part, la SVD, qui est classiquement définie a posteriori, peut également être définie a priori en choisissant une norme adéquate pour minimiser le résidu (et de même pour la POD). On verra dans le chapitre suivant comment tirer parti de cette définition.

À la question quelle définition utiliser pour calculer la PGD, on pourra répondre de tester toutes les définitions et de choisir la plus efficace (c'est ce que l'on fera à la fin de ce chapitre pour des problèmes de propagation d'ondes). En effet, la convergence de la PGD a été prouvée pour des problèmes symétriques avec la définition (G)PGD [Le Bris *et al.*, 2009, Cancès *et al.*, 2011, Falco et Nouy, 2011] et pour des problèmes non-symétriques avec la définition (R)PGD [Ammar *et al.*, 2010b, Falcó et Nouy, 2012, Cancès *et al.*, 2012]. Cependant, la convergence de la définition (G)PGD a été observée numériquement pour des problèmes non-symétriques (voir les résultats obtenus par [Ammar *et al.*, 2007, Nouy, 2010a] dans le cas de problèmes d'évolution de type parabolique). La définition (R)PGD demande, à chaque itération, plus d'effort numérique que la définition (G)PGD. Aussi, on privilégiera la PGD avec critère de Galerkin (lorsqu'elle converge) dans le cas d'un problème non-symétrique. Enfin, la définition avec critère de Petrov-Galerkin a été proposée dans le but de résoudre des problèmes non-symétriques (il n'y a pas de résultat général de convergence pour cette définition). Cette définition a conduit à une approximation plus précise (que celles données par les définitions (G)PGD et (R)PGD) dans les cas qui ont été testés par [Nouy, 2010a].

Remarque 5.5. *Dans le cas de tenseurs d'ordre élevé ($D\geq 3$), la PGD consiste à chercher une approximation de rang M dans le sous-ensemble* R^D_M *des décompositions canon-*

iques de rang M, défini par

$$\mathrm{R}_M^D = \left\{ \mathbf{u} \in \bigotimes_{d=1}^{D} \mathbb{R}^{n_d} \mid \mathbf{u} = \sum_{m=1}^{M} \bigotimes_{d=1}^{D} \boldsymbol{w}_m^d \quad avec \quad \boldsymbol{w}_m^d \in \mathbb{R}^{n_d} \right\}. \tag{5.14}$$

Dans le cas de décomposition a posteriori, ce format de tenseur a été introduit par [Carroll et Chang, 1970, Harshman, 1970] dans le but d'approcher un tenseur $\mathbf{u} \in \bigotimes_{d=1}^{D} \mathbb{R}^{n_d}$ *(connu), par un tenseur de faible rang cherché dans* R_M^D*, de façon très similaire à la PGD*[7]*. Cependant, dans le cas où* $D \geq 3$ *et* $M > 1$ *cet ensemble n'est pas fermé, et le problème de meilleure approximation dans* R_M^D *n'a pas nécessairement de solution [De Silva et Lim, 2008]. En pratique, ceci se traduit par la non convergence de l'algorithme ALS dans de nombreux cas*[8]*. Différentes stratégies ont été proposées pour s'affranchir de cette difficulté [Acar* et al.*, 2011, Espig* et al.*, 2012, De Sterck, 2012, De Sterck et Miller, 2013]. Dans le cadre de la PGD, cette difficulté est surmontée en exploitant une stratégie gloutonne (voir la suite de cette section) où le problème de minimisation*[9] *est (bien) posé dans* R_1^D*. La démonstration de la convergence de cette stratégie est notamment l'objet des publications déjà mentionnées de [Le Bris* et al.*, 2009, Ammar* et al.*, 2010b, Cancès* et al.*, 2011, Falco et Nouy, 2011, Falcó et Nouy, 2012, Cancès* et al.*, 2012]. Une alternative, proposée dans la communauté a posteriori, est de changer de format de tenseurs (voir la revue de [Kolda et Bader, 2009] et l'ouvrage de [Hackbusch, 2012] pour un bestiaire des formats les plus récents). De cette façon, on peut définir un problème de meilleure approximation « bien posé » dans un sous-ensemble fermé de tenseurs de rang M (et éviter les difficultés numériques rencontrées sinon). Parmis les plus populaires, on citera les formats de Tucker et Tucker hiérarchique [Hackbusch et Kühn, 2009]. On pourra consulter la revue récente de [Grasedyck* et al.*, 2013] sur les techniques de construction associées à une approximation de faible rang dans ce format. Dans la communauté a priori, de tels formats de tenseurs ont été couplés aux méthodes classiques de projection [Kressner et Tobler, 2010, Khoromskij et Schwab, 2011, Matthies et Zander, 2012, Ballani et Grasedyck, 2013]. Les formats de Tucker et Tucker hiérarchique ont été introduits, dans le cadre d'approximations de type PGD, par [Giraldi, 2012].*

7. La meilleure approximation dans R_M^D est définie a posteriori au sens d'un problème de minimisation. Un algorithme de minimisation alternée, appelé « Alternative Least Square (ALS) », est introduit pour calculer la décomposition. La minimisation est effectuée alternativement sur tous les modes dans une dimension donnée.

8. L'algorithme peut par exemple converger pour un rang donné mais pas pour un autre rang inférieur ou supérieur.

9. La stratégie gloutonne ne permet cependant pas d'obtenir un minimum global. Aussi des stratégies de mise à jour doivent être employées pour améliorer la précision de l'approximation [Nouy, 2010a, Falcó et Nouy, 2012].

5.3 Algorithmes

Dans cette section, on présente les prototypes des algorithmes classiquement utilisés pour résoudre le problème de minimisation (5.13) associé aux définitions classiques [10] de la PGD. Globalement, on peut distinguer deux types de stratégie, à savoir la construction directe et la construction gloutonne, pour lesquelles le problème (5.13) est défini respectivement dans R_M ou R_1.

Remarque 5.6. *Pour ces deux types de construction, le problème (5.13) est résolu à l'aide d'une stratégie de minimisation alternée. Les mappings définis dans le cadre de cette stratégie, sont détaillés pour la PGD associée au critère d'orthogonalité de Galerkin. La PGD en minimum de résidu (dans la norme* $\|\ \|_2$*) peut être implémentée à partir de ces mappings en remplaçant simplement les opérateurs* $\mathbf{A}$ *et* $\mathbf{b}$ *du problème par les opérateurs* $\mathbf{A}'\,\dot{}^{D}\mathbf{A}$ *et* $\mathbf{A}'\,\dot{}^{D}\mathbf{b}$ *associés au problème symétrisé (à gauche). On pourra en effet vérifier l'égalité suivante pour tout* $\mathbf{u}^\star \in \mathrm{R}_M$*,*

$$\frac{\delta(\frac{1}{2}\left\|\mathbf{b}-\mathbf{A}\,\dot{}^{D}\mathbf{u}^\star\right\|_2^2)}{\delta\mathbf{u}^\star}\,\dot{}^{D}\delta\mathbf{u}^\star = \frac{\delta(\frac{1}{2}\langle\mathbf{u}^\star,\mathbf{u}^\star\rangle_{\mathbf{A}'\,\dot{}^{D}\mathbf{A}}-\langle\mathbf{u}^\star,\mathbf{A}'\,\dot{}^{D}\mathbf{b}\rangle_2)}{\delta\mathbf{u}^\star}\,\dot{}^{D}\delta\mathbf{u}^\star\,. \tag{5.15}$$

De façon équivalente, les mappings associés au critère d'orthogonalité de Petrov-Galerkin peuvent être appliqués directement aux opérateurs du problème symétrisé.

5.3.1 Construction directe

La construction directe consiste à minimiser la fonctionnelle J par rapport à tous les modes, en alternant la minimisation par rapport aux modes spatiaux et temporels. Cette minimisation alternée s'écrit :

- connaissant $[\boldsymbol{\lambda}_1,\cdots,\boldsymbol{\lambda}_M]\in\mathbb{R}^{n_T\times M}$, trouver $[\boldsymbol{w}_1,\cdots,\boldsymbol{w}_M]\in\mathbb{R}^{n_S\times M}$ tels que :

$$[\boldsymbol{w}_1,\cdots,\boldsymbol{w}_M] = \arg\min_{[\boldsymbol{w}_1^\star,\cdots,\boldsymbol{w}_M^\star]\in\mathbb{R}^{n_S\times M}} J(\sum_{m=1}^{M}\boldsymbol{w}_m^\star\otimes\boldsymbol{\lambda}_m)\,, \tag{5.16a}$$

- connaissant $[\boldsymbol{w}_1,\cdots,\boldsymbol{w}_M]\in\mathbb{R}^{n_S\times M}$, trouver $[\boldsymbol{\lambda}_1,\cdots,\boldsymbol{\lambda}_M]\in\mathbb{R}^{n_T\times M}$ tels que :

$$[\boldsymbol{\lambda}_1,\cdots,\boldsymbol{\lambda}_M]\in\mathbb{R}^{n_T\times M} = \arg\min_{[\boldsymbol{\lambda}_1^\star,\cdots,\boldsymbol{\lambda}_M^\star]\in\mathbb{R}^{n_T\times M}} J(\sum_{m=1}^{M}\boldsymbol{w}_m\otimes\boldsymbol{\lambda}_m^\star)\,. \tag{5.16b}$$

On répète alors successivement ces deux étapes jusqu'à convergence de la méthode. Chaque étape de (5.16) est associée à la résolution d'un système linéaire. Celui-ci est obtenu en écrivant la condition de stationnarité associée au problème de minimisa-

10. L'implémentation de la définition avec critère de Petrov-Galerkin n'est pas détaillée. On pourra consulter [Nouy, 2010a] pour les détails concernant la définition des mappings.

tion. Pour la définition (G)PGD, les conditions de stationnarité associées aux problèmes de minimisation (5.16a) et (5.16b) aboutissent aux mappings suivants :

$$S^M_{\text{direct}} : \begin{cases} \mathbb{R}^{n_T \times M} \to \mathbb{R}^{n_S \times M} \\ [\boldsymbol{\lambda}_1, \cdots, \boldsymbol{\lambda}_M] \mapsto [\boldsymbol{w}_1, \cdots, \boldsymbol{w}_M] \end{cases} \quad \text{tel que} \quad \begin{bmatrix} \mathbf{S}_{11} & - & \mathbf{S}_{1M} \\ | & \diagdown & | \\ \mathbf{S}_{M1} & - & \mathbf{S}_{MM} \end{bmatrix} . \begin{bmatrix} \boldsymbol{w}_1 \\ | \\ \boldsymbol{w}_M \end{bmatrix} = \begin{bmatrix} \boldsymbol{f}^S_1 \\ | \\ \boldsymbol{f}^S_M \end{bmatrix}, \quad (5.17a)$$

$$T^M_{\text{direct}} : \begin{cases} \mathbb{R}^{n_S \times M} \to \mathbb{R}^{n_T \times M} \\ [\boldsymbol{w}_1, \cdots, \boldsymbol{w}_M] \mapsto [\boldsymbol{\lambda}_1, \cdots, \boldsymbol{\lambda}_M] \end{cases} \quad \text{tel que} \quad \begin{bmatrix} \mathbf{T}_{11} & - & \mathbf{T}_{1M} \\ | & \diagdown & | \\ \mathbf{T}_{M1} & - & \mathbf{T}_{MM} \end{bmatrix} . \begin{bmatrix} \boldsymbol{\lambda}_1 \\ | \\ \boldsymbol{\lambda}_M \end{bmatrix} = \begin{bmatrix} \boldsymbol{f}^T_1 \\ | \\ \boldsymbol{f}^T_M \end{bmatrix}, \quad (5.17b)$$

où les opérateurs $\mathbf{S}_{mn}$ et $\mathbf{T}_{mn}$ sont donnés pour $m = 1, \cdots, M$ et $n = 1, \cdots, M$ par

$$\mathbf{S}_{mn} = \sum_{k=1}^{M_A} (\boldsymbol{\lambda}_m . \mathbf{A}^T_k . \boldsymbol{\lambda}_n) \mathbf{A}^S_k \quad \text{et} \quad \mathbf{T}_{mn} = \sum_{k=1}^{M_A} (\boldsymbol{w}_m . \mathbf{A}^S_k . \boldsymbol{w}_n) \mathbf{A}^T_k , \quad (5.17c)$$

et les vecteurs $\boldsymbol{f}^S_m$ et $\boldsymbol{f}^T_m$ sont donnés pour $m = 1, \cdots, M$ par

$$\boldsymbol{f}^S_m = \sum_{k=1}^{M_b} (\boldsymbol{\lambda}_m . \boldsymbol{b}^T_k) \boldsymbol{b}^S_k \quad \text{et} \quad \boldsymbol{f}^T_m = \sum_{k=1}^{M_b} (\boldsymbol{w}_m . \boldsymbol{b}^S_k) \boldsymbol{b}^T_k . \quad (5.17d)$$

La construction directe est résumée dans l'Algorithme 1. Cet algorithme permet de trouver la solution optimale (qui minimise la fonctionnelle par rapport à tous les modes) du problème de meilleure approximation (5.13). Cependant, il nécessite de résoudre des problèmes spatiaux et temporels dont la taille (respectivement donné par $M n_S \times M n_S$ et $M n_S \times M n_T$) dépend du rang M de l'approximation. Aussi, bien que cet algorithme donne la meilleure approximation de rang M (au sens de la norme de l'opérateur), il est inutilisable en pratique dès lors que le rang M devient trop important.

Algorithm 1 Construction directe (PGD-S)

Entrées : $\mathbf{A} = \sum_{k=1}^{M_A} \mathbf{A}^S_k \otimes \mathbf{A}^T_k$, $\mathbf{b} = \sum_{k=1}^{M_b} \boldsymbol{b}^S_k \otimes \boldsymbol{b}^T_k$

Sortie : $\mathbf{u} = \sum_{m=1}^{M} \boldsymbol{w}_m \otimes \boldsymbol{\lambda}_m$

Paramètres : $M, \xi_{\max}, \epsilon_{\max}$

1: Initialiser $[\boldsymbol{\lambda}_1, \cdots, \boldsymbol{\lambda}_M]$
2: **for** $\xi = 1$ to $\xi_{\max}$
3: $\quad [\boldsymbol{w}_1, \cdots, \boldsymbol{w}_M] = S^M_{\text{direct}}([\boldsymbol{\lambda}_1, \cdots, \boldsymbol{\lambda}_M])$
4: $\quad [\boldsymbol{\lambda}_1, \cdots, \boldsymbol{\lambda}_M] = T^M_{\text{direct}}([\boldsymbol{w}_1, \cdots, \boldsymbol{w}_M])$
5: $\quad$ Normaliser
6: $\quad$ Vérifier la convergence par rapport à $\epsilon_{\max}$
7: **end**

5.3.2 Constructions gloutonnes

La construction la plus simple et la moins coûteuse de la PGD est basée sur un algorithme glouton (« greedy algorithm »). Cette construction est résumée dans l'Algorithme 2. Pour construire l'approximation de rang m, on suppose que l'on connait l'approximation de rang $m-1$, puis on calcule le meilleure enrichissement de rang un, noté $\boldsymbol{w}\otimes\boldsymbol{\lambda}$, qui minimise le résidu courant. C'est-à-dire, connaissant $\mathbf{u}_{m-1}$, on cherche $\boldsymbol{w}\otimes\boldsymbol{\lambda}\in \mathrm{R}_1$ tel que

$$\boldsymbol{w}\otimes\lambda = \arg\min_{\mathbf{u}^\star\in \mathrm{R}_1} J(\mathbf{u}_{m-1}+\mathbf{u}^\star)\,. \tag{5.18}$$

Un fois cet enrichissement calculé, on met à jour la décomposition courante, c'est-à-dire $\mathbf{u}_m = \mathbf{u}_{m-1}+\boldsymbol{w}\otimes\lambda$, puis on incrémente le rang m jusqu'à obtenir le décomposition de rang M cherchée.

Le problème (5.18) de minimisation dans R_1 est résolue avec la stratégie de minimisation alternée [11]. En écrivant la condition de stationnarité associée au problème (5.18) alternativement par rapport aux modes spatial et temporel, on aboutit aux mappings suivants :

$$S_{\text{glouton}} : \begin{cases} \mathbb{R}^{n_T} \to \mathbb{R}^{n_S} \\ \boldsymbol{\lambda} \mapsto \boldsymbol{w} \end{cases} \quad \text{tel que} \quad \mathbf{S}.\boldsymbol{w} = \boldsymbol{f}^S\,, \tag{5.19a}$$

$$T_{\text{glouton}} : \begin{cases} \mathbb{R}^{n_S} \to \mathbb{R}^{n_T} \\ \boldsymbol{w} \mapsto \boldsymbol{\lambda} \end{cases} \quad \text{tel que} \quad \mathbf{T}.\boldsymbol{\lambda} = \boldsymbol{f}^T\,, \tag{5.19b}$$

où les opérateurs $\mathbf{S}$ et $\mathbf{T}$ sont donnés par

$$\mathbf{S} = \sum_{k=1}^{M_A} (\boldsymbol{\lambda}.\mathbf{A}_k^T.\boldsymbol{\lambda})\mathbf{A}_k^S \quad \text{et} \quad \mathbf{T} = \sum_{k=1}^{M_A} (\boldsymbol{w}.\mathbf{A}_k^S.\boldsymbol{w})\mathbf{A}_k^T\,, \tag{5.19c}$$

et les vecteurs $\boldsymbol{f}^S$ et $\boldsymbol{f}^T$ sont donnés en fonction du résidu $\mathbf{r} = \sum_{k=1}^{M_r} \boldsymbol{r}_k^S \otimes \boldsymbol{r}_k^T$ par

$$\boldsymbol{f}^S = \sum_{k=1}^{M_r} (\boldsymbol{\lambda}.\boldsymbol{r}_k^T)\boldsymbol{r}_k^S \quad \text{et} \quad \boldsymbol{f}^T = \sum_{k=1}^{M_r} (\boldsymbol{w}.\boldsymbol{r}_k^S)\boldsymbol{r}_k^T\,. \tag{5.19d}$$

Remarque 5.7. *On remarquera dans l'Algorithme 2 que les étapes de résolution des mappings (étapes (5) et (6)) dépendent du résidu courant* $\mathbf{r}$*, qui est actualisé à l'étape (10). Aussi, cette étape d'actualisation du résidu doit être regardée avec précautions. En effet, le rang du résidu courant (noté* M_r*) augmente à chaque fois que cette étape est réalisée. Le sous-ensemble* R_{M_r} *des décompositions canoniques de rang* M_r *n'étant pas un espace vectoriel, on ne peut pas « écraser » la variable utilisée pour le stockage du*

11. On fixe $\boldsymbol{\lambda}$ et on minimise dans $\mathbb{R}^{n_S}\otimes\{\boldsymbol{\lambda}\}$ pour obtenir $\boldsymbol{w}$, puis on fixe $\boldsymbol{w}$ et on minimise dans $\{\boldsymbol{w}\}\otimes\mathbb{R}^{n_T}$ pour obtenir $\boldsymbol{\lambda}$, et ainsi de suite.

résidu par sa valeur actualisée. Aussi, la somme dans l'étape (10) doit être vue comme une étape de stockage du résidu sous la forme $\sum_{k=1}^{M_r} \boldsymbol{r}_k^S \otimes \boldsymbol{r}_k^T + \sum_{k=1}^{M_A} \mathbf{A}_k^S.\boldsymbol{w} \otimes \mathbf{A}_k^T.\boldsymbol{\lambda}$*. Cette stratégie permet de calculer une fois pour toute, les (nombreux) produits matrice-vecteur que l'on rencontre au moment de l'assemblage des mappings des étapes (5) et (6). Le rang du résidu à l'itération m est cependant donné par* $M_r = M_b + (m-1)M_A$ *et son stockage peut rapidement devenir coûteux si m devient trop grand.*

Algorithm 2 Construction gloutonne (PGD-P)

Entrées : $\mathbf{A} = \sum_{k=1}^{M_A} \mathbf{A}_k^S \otimes \mathbf{A}_k^T$, $\mathbf{b} = \sum_{k=1}^{M_b} \boldsymbol{b}_k^S \otimes \boldsymbol{b}_k^T$

Sortie : $\mathbf{u} = \sum_{m=1}^{M} \boldsymbol{w}_m \otimes \boldsymbol{\lambda}_m$

Paramètres : $M, \xi_{\max}, \epsilon_{\max}$

1: $\mathbf{r} = \mathbf{b}$
2: **for** $m = 1$ to M
3: Initialiser $\boldsymbol{\lambda}$
4: **for** $\xi = 1$ to $\xi_{\max}$
5: $\boldsymbol{w} = S_{\text{glouton}}(\boldsymbol{\lambda}; \mathbf{r})$
6: $\boldsymbol{\lambda} = T_{\text{glouton}}(\boldsymbol{w}; \mathbf{r})$
7: Normaliser
8: Vérifier la convergence par rapport à $\epsilon_{\max}$
9: **end**
10: $\mathbf{r} = \mathbf{r} - \mathbf{A}^{\mathrm{D}}.\boldsymbol{w} \otimes \boldsymbol{\lambda}$
11: $\boldsymbol{w}_m = \boldsymbol{w}$, $\boldsymbol{\lambda}_m = \boldsymbol{\lambda}$
12: **end**

Construction gloutonne avec mises à jour

L'approximation de rang M obtenue à la fin de la construction gloutonne n'est pas optimale (on minimise seulement le problème (5.13) par rapport au mode courant). Aussi, des stratégies de mise à jour de toute la décomposition courante, ont été proposées afin d'améliorer la précision de l'approximation gloutonne. La plus part des stratégies proposées [12] consiste à effectuer quelques résolutions alternatives des mappings S_{direct}^m et T_{direct}^m en les initialisants avec la décomposition courante de rang m, avant de passer au calcul du mode suivant [Nouy, 2010a, Falcó et Nouy, 2012]. Cette stratégie nécessite cependant de pouvoir résoudre ces mappings, ce qui n'est plus possible dès lors que m devient trop grand. On propose ici une approche légèrement différente qui permet de mettre à jour les modes même lorsque m devient grand

12. Une autre approche consiste à exploiter les itérés du processus de minimisations alternées (donnés par $\boldsymbol{w}^{(0)}$ et $\boldsymbol{w}^{(\xi)} = S^1_{\text{glouton}}(T^1_{\text{glouton}}(\boldsymbol{w}^{(\xi-1)}))$ pour $\xi = 1, \cdots, M$) pour constuire un sous-espace de modes spatiaux (qui peut être vue comme un sous-espace de Krylov), puis de mettre à jour les modes temporels sur ce sous-espace avec le mapping T_{direct}^M [Nouy, 2008, Tamellini *et al.*, 2012].

[Boucinha *et al.*, 2013b]. L'idée est d'effectuer ξ_{udp} itérations d'un algorithme de minimisations alternées dans R_m, qui peut être vue comme une approche de type Gauss-Seidel par blocs [13] pour résoudre les mappings S^m_{direct} et T^m_{direct}. La construction gloutonne avec cette stratégie de mise à jour est résumée dans l'Algorithme 3.

Algorithm 3 Construction gloutonne avec mise à jour (PGD-P+upd)

Entrées : $\mathbf{A} = \sum_{k=1}^{M_A} \mathbf{A}_k^S \otimes \mathbf{A}_k^T$, $\mathbf{b} = \sum_{k=1}^{M_b} \boldsymbol{b}_k^S \otimes \boldsymbol{b}_k^T$

Sortie : $\mathbf{u} = \sum_{m=1}^{M} \boldsymbol{w}_m \otimes \boldsymbol{\lambda}_m$

Paramètres : $M, \xi_{\max}, \epsilon_{\max}, \xi_{\text{upd}}$

1: $\mathbf{r} = \mathbf{b}$
2: **for** $m = 1$ to M
3: Faire les étapes (3) à (11) de l'Algorithme 2
4: **for** $\xi = 1$ to ξ_{upd}
5: **for** $i = 1$ to m
6: $\mathbf{r} = \mathbf{b} - \sum_{j=1, j\neq i}^{m} \mathbf{A}^{\text{D}} \boldsymbol{w}_j \otimes \boldsymbol{\lambda}_j$
7: $\boldsymbol{w}_i = S_{\text{glouton}}(\boldsymbol{\lambda}_i; \mathbf{r})$
8: $\boldsymbol{\lambda}_i = T_{\text{glouton}}(\boldsymbol{w}_i; \mathbf{r})$
9: Normaliser
10: **end**
11: **end**
12: $\mathbf{r} = \mathbf{b} - \sum_{i=1}^{m} \mathbf{A}^{\text{D}} \boldsymbol{w}_i \otimes \boldsymbol{\lambda}_i$
13: **end**

Remarque 5.8. *On remarquera dans l'Algorithme 3, que la boucle de mise à jour de la décomposition courante (étape (4)) nécessite de re-calculer le résidu courant. De ce fait, l'assemblage des mappings (7) et (8) est plus coûteux que dans le cas d'une construction gloutonne pure.*

Construction gloutonne avec projection

Dans l'approche initiale proposée par [Ammar *et al.*, 2007], une étape de projection sur une base réduite est ajoutée à l'algorithme glouton. Supposons que l'on connaisse M modes espace-temps $\boldsymbol{w}_1 \otimes \boldsymbol{\lambda}_1, \cdots, \boldsymbol{w}_M \otimes \boldsymbol{\lambda}_M$, et que l'on puisse construire un espace réduit [14] (noté U_M) à partir de ces modes. Dans ce cas la projection $\mathbf{u}_M$ de $\mathbf{u}$ sur l'espace réduit U_M s'écrit sous la forme,

$$\mathbf{u}_M = \sum_{m=1}^{M} \alpha_m \boldsymbol{w}_m \otimes \boldsymbol{\lambda}_m \quad \text{avec} \quad \alpha_m \in \mathbb{R}, \tag{5.20}$$

13. On écrit le problème de minimisation dans R_m mode par mode, en alternant l'espace et le temps.
14. On suppose que $\dim(\text{U}_M) = M << n_S n_T$

et les coefficients α_m sont obtenus en résolvant le système linéaire suivant :

$$\sum_{n=1}^{M} \langle \boldsymbol{w}_m \otimes \boldsymbol{\lambda}_m , \boldsymbol{w}_n \otimes \boldsymbol{\lambda}_n \rangle_{\mathbf{A}} \alpha_n = \langle \boldsymbol{w}_m \otimes \boldsymbol{\lambda}_m , \mathbf{b} \rangle_2, \quad \text{pour } m = 1, \cdots, M. \tag{5.21}$$

En pratique, on observe peu de différence entre l'approximation de rang M construite de cette façon et celle obtenue avec une construction gloutonne pure. Une amélioration de cette stratégie a été proposée par [Giraldi, 2012]. L'idée est de construire un espace réduit « plus riche ». On suppose cette fois que l'on connait M modes en espace et M modes en temps, et on définit deux espaces réduits à partir de ces modes, notés respectivement U_M^S et U_M^T. L'espace réduit espace-temps (noté U_M^{ST}) est alors construit par tensorisation de ces espaces, soit $\mathrm{U}_M^{ST} = \mathrm{U}_M^S \otimes \mathrm{U}_M^T$. Dans ce cas, la projection $\mathbf{u}_M$ de $\mathbf{u}$ sur U_M^{ST} s'écrit sous la forme,

$$\mathbf{u}_M = \sum_{m_S=1}^{M} \sum_{m_T=1}^{M} \alpha_{m_S m_T} \boldsymbol{w}_{m_S} \otimes \boldsymbol{\lambda}_{m_T}, \tag{5.22}$$

et les coefficients $\alpha_{m_S m_T}$ sont obtenus en résolvant le système linéaire

$$\sum_{n_S=1}^{M} \sum_{n_T=1}^{M} \langle \boldsymbol{w}_{m_S} \otimes \boldsymbol{\lambda}_{m_T} , \boldsymbol{w}_{n_S} \otimes \boldsymbol{\lambda}_{n_T} \rangle_{\mathbf{A}} \alpha_{n_S n_T} = \langle \boldsymbol{w}_{m_S} \otimes \boldsymbol{\lambda}_{m_T} , \mathbf{b} \rangle_2, \tag{5.23}$$
$$\text{pour } m_S = 1, \cdots, M, \text{ et } m_T = 1, \cdots, M.$$

L'utilisation de cette projection dans le cadre d'une construction gloutonne de la base réduite est résumée dans l'Algorithme 4. Les étapes (4) et (5) nécessitent de définir un produit scalaire pour orthogonaliser les modes. On choisit ici le produit scalaire canonique.

Remarque 5.9. *En pratique, on stoppe l'algorithme à l'étape (4) ou (5) si rang*$(\mathrm{U}_{m-1}^S) =$ *rang*(U_m^S) *ou rang*$(\mathrm{U}_{m-1}^T) =$ *rang*(U_m^T).

Remarque 5.10. *Le système linéaire associé à la projection peut rapidement devenir coûteux lorsque le rang m augmente. On doit en effet résoudre un système linéaire de taille* $m^2 \times m^2$*, dont la matrice est entièrement peuplée.*

Remarque 5.11. *La décomposition écrite sous la forme de l'équation (5.22) appartient au sous-ensemble des tenseurs de Tucker. L'utilisation de ce format dans le cadre des algorithmes PGD nécessite une implémentation spécifique. On notera que des implémentations Matlab pour l'approximation a posteriori de tenseur dans ce type de format (Tucker, Tucker hiérarchique,...) ont notamment été proposées par [Bader* et al.*, 2012].*

Algorithm 4 Construction gloutonne avec projection (PGD-P+proj)

Entrées : $\mathbf{A} = \sum_{k=1}^{M_A} \mathbf{A}_k^S \otimes \mathbf{A}_k^T$, $\mathbf{b} = \sum_{k=1}^{M_b} \boldsymbol{b}_k^S \otimes \boldsymbol{b}_k^T$

Sortie : $\mathbf{u} = \sum_{i=1}^{M} \sum_{j=1}^{M} \alpha_{ij} \boldsymbol{w}_i \otimes \boldsymbol{\lambda}_j$

Paramètres : $M, \xi_{\max}, \epsilon_{\max}$

1: $\mathbf{r} = \mathbf{b}$
2: **for** $m = 1$ to M
3: Faire les étapes (3) à (11) de l'Algorithme 2
4: Orthonormaliser $\boldsymbol{w}_m$ à $\boldsymbol{w}_1, \cdots, \boldsymbol{w}_{m-1}$
5: Orthonormaliser $\boldsymbol{\lambda}_m$ à $\boldsymbol{\lambda}_1, \cdots, \boldsymbol{\lambda}_{m-1}$
6: Résoudre la projection (5.23)
7: $\mathbf{r} = \mathbf{r} - \mathbf{A}^{\mathrm{D}} \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_{ij} \boldsymbol{w}_i \otimes \boldsymbol{\lambda}_j$
8: **end**

5.3.3 Commentaires

Certains détails techniques n'ont pas été mentionnés dans la description des algorithmes. Ces points concernent l'initialisation des modes, le critère d'arrêt pour les itérations de minimisations alternées et la normalisation des modes. Ils sont rapidement précisés dans cette partie. Pour simplifier la lecture, on introduit les acronymes suivants de façon similaire à ceux proposés par [Nouy, 2010a] :

- PGD-S désigne l'algorithme de construction directe [15],
- PGD-P désigne l'algorithme de construction gloutonne pure [16],
- PGD-P+upd désigne l'algorithme de construction gloutonne avec mise à jour,
- PGD-P+proj désigne l'algorithme de construction gloutonne avec projection.

Initialisation des modes

Pour tous les algorithmes gloutons, l'initialisation du mode en temps est réalisée avec un vecteur peuplé de uns, que l'on normalise. L'initialisation pour l'algorithme PGD-S doit être effectuée avec plus de précautions. On peut par exemple utiliser un des algorithmes gloutons pour calculer une approximation de rang M qui serve d'initialisation pour les modes en temps. Cependant, les modes en temps obtenus de cette façon peuvent être linéairement dépendants, et l'algorithme ne converge pas dans ce cas. Aussi, la stratégie la plus robuste est d'initialiser l'algorithme PGD-S en procédant de façon progressive : on calcule u_1, u_2 jusqu'à u_M (avec l'algorithme PGD-S à chaque fois) en initialisant la décomposition de rang m avec la décomposition de rang $m-1$, le dernier mode étant initialisé avec un vecteur de uns normalisé.

15. Cet algorithme peut être vue comme un algorithme d'itérations du sous-espace, d'où le S pour « Subspace iterations ».

16. Cet algorithme peut être vue comme un algorithme de puissances itérées avec déflation, d'où le P pour « Power iterations ».

Critère d'arrêt pour les itérations de minimisations alternées

Pour tous les algorithmes, le critère d'arrêt pour les itérations de minimisations alternées, est basé sur la stagnation de la fonctionnelle que l'on cherche à minimiser. Ce critère s'écrit à l'itération ξ de la façon suivante :

$$\epsilon(\xi) = \frac{|J^{(\xi)} - J^{(\xi-1)}|}{J^{(\xi)}}, \tag{5.24}$$

$$\text{avec} \quad J^{(\xi)} = \begin{cases} J(\mathbf{u}_M^{(\xi)}) & \text{(PGD-S)} \\ J(\mathbf{u}_{m-1} + \boldsymbol{w}^{(\xi)} \otimes \boldsymbol{\lambda}^{(\xi)}) & \text{(PGD-P)} \end{cases}.$$

On stoppe les itérations de minimisations alternées lorsque $\epsilon(\xi) \leq \epsilon_{\max}$.

Normalisation des modes

La normalisation des modes, entre les étapes du processus de minimisations alternées, permet d'éviter un phénomène rencontré assez fréquemment lorsque les modes ne sont pas normalisés. Dans ce cas, on observe que la norme du mode spatial tend vers zéros alors que la norme du mode en temps tend vers l'infini (ou inversement). Pour éviter ce phénomène, on normalise les modes (spatiaux et temporels) après chaque résolution d'un mapping. Puis à la fin d'une itération donnée, on redistribue la norme du mode espace-temps sur chaque mode en espace et en temps [17]. C'est-à-dire, que pour chaque itération de minimisations alternées, on réalise les étapes suivantes :

1: $\boldsymbol{w} = S(\boldsymbol{\lambda})$
2: $\alpha^S = \| \boldsymbol{w} \|_2$
3: $\boldsymbol{w} = \boldsymbol{w} / \alpha^S$
4: $\boldsymbol{\lambda} = T(\boldsymbol{w})$
5: $\alpha^T = \| \boldsymbol{\lambda} \|_2$
6: $\boldsymbol{\lambda} = \boldsymbol{\lambda} / \alpha^T$
7: $\alpha = \alpha^T \alpha^S$
8: $\boldsymbol{w} = \sqrt{\alpha} \boldsymbol{w}$
9: $\boldsymbol{\lambda} = \sqrt{\alpha} \boldsymbol{\lambda}$

Remarque 5.12. *Le choix de la norme $\| \ \|_2$ est purement pratique (c'est la normalisation la moins coûteuse). Cependant, il peut être intéressant de normaliser les modes espace-temps (à convergence du processus de minimisation alternées) dans la norme associée à la définition a posteriori de la décomposition. Dans ce cas, le scalaire α peut être utilisé pour évaluer la précision de l'approximation de rang M obtenue. On a en effet la propriété suivante pour toutes les définitions a posteriori du Tableau 5.1,*

$$\| \mathbf{u} - \mathbf{u}_M \|^2 = \| \mathbf{u} \|^2 - \sum_{m=1}^{M} \alpha_m^2, \tag{5.25}$$

17. Cette procédure est notamment utilisée dans la toolbox Matlab de [Bader *et al.*, 2012].

où le scalaire α_m est la norme du $m^{ième}$ mode espace-temps à la sortie du processus de minimisation alternée, et $\|\ \|$ est la norme de la définition a posteriori. Une stratégie grossière pour choisir le rang M de l'approximation de façon adaptative, est ainsi de comparer les valeurs de α_m.

Exemple 5.1. (Complexité des algorithmes PGD) Dans cet exemple, on évalue la complexité des algorithmes PGD. Pour tous les algorithmes, l'étape la plus coûteuse est la résolution des mappings.

Dans le Tableau 5.2, on a regroupé la complexité de cette étape, on supposant que l'on cherche une approximation de rang M et que le processus de minimisations alternées converge en $\xi_{\max}$ itérations. On rappelle que $\mathbf{lin}(n)$ est la complexité associée à la résolution d'un système linéaire de taille $n \times n$. Pour les mêmes valeurs de M et $\xi_{\max}$, les constructions gloutonnes pure et avec projection sont les moins coûteuses. La mise à jour de la contruction gloutonne ajoute un coût de l'ordre de $\frac{M^2}{2}\xi_{\text{upd}}(\mathbf{lin}(n_S)+\mathbf{lin}(n_T))$ qui n'est donc pas négligeable lorsque M est grand. Enfin, le coût de la construction directe devient rapidement prohibitif lorsque le rang M augmente (les systèmes linéaires à résoudre sont de tailles $Mn_S \times Mn_S$ ou $Mn_T \times Mn_T$).

Algorithme	Complexité de la résolution des mappings
PGD-P	$M\xi_{\max}\big(\mathbf{lin}(n_S)+\mathbf{lin}(n_T)\big)$
PGD-P+proj	$M\xi_{\max}\big(\mathbf{lin}(n_S)+\mathbf{lin}(n_T)\big)$
PGD-P+upd	$\big(M\xi_{\max}+\frac{M(M+1)}{2}\xi_{\text{upd}}\big)\big(\mathbf{lin}(n_S)+\mathbf{lin}(n_T)\big)$
PGD-S	$\xi_{\max}\big(\mathbf{lin}(Mn_S)+\mathbf{lin}(Mn_T)\big)$

TABLE 5.2: Complexité associé à la résolution des mappings pour différents algorithmes PGD.

À la complexité de la résolution des mappings, on doit également ajouter la complexité de leur assemblage. Les opérations associées sont des produits matrice-vecteur (on compte $(2\text{nnz}-n)$ opérations), des produits scalaires (on compte n opérations) et des sommes de matrices (on compte nnz opérations), où on a noté $\text{nnz} = \text{nnz}_S + \text{nnz}_T$ avec nnz_S et nnz_T les nombres moyens d'entrées différentes de zéros dans les opérateurs $\mathbf{A}_k^S$ et $\mathbf{A}_k^T$ respectivement, et $n = n_S + n_T$ avec n_S et n_T les dimensions des espaces d'approximation. En supposant que l'on stocke le résidu courant sous la forme décrite dans la Remarque 5.7, on obtient les complexités présentées dans le Tableau 5.3 pour les constructions gloutonne pure et directe.

Algorithme	Complexité de l'assemblage des mappings
PGD-P	$\xi_{\max}(3MM_A\text{nnz} + M(M-1)M_An + 2MM_bn) + MM_A(2\text{nnz} - n)$
PGD-S	$\xi_{\max}(3M^2M_A\text{nnz} + 2MM_bn)$

TABLE 5.3: Complexité associé à l'assemblage des mappings pour différents algorithmes PGD.

En supposant que nnz >> n, l'assemblage des mappings nécessite de l'ordre de $3\xi_{\max}MM_A$nnz pour l'algorithme PGD-P et de l'ordre de $3\xi_{\max}M^2M_A$nnz pour l'algorithme PGD-S. On comprend donc que cette étape n'est pas négligeable dès lors que $\xi_{\max}$, M ou M_A sont grands. Notamment pour le problème symétrisé, on a $M_{A'A} = M_A^2$. Pour cette raison, la définition PGD en minimum de résidu est plus coûteuse que la définition avec critère de Galerkin. On notera cependant que toutes ces opérations peuvent être parallélisées avec un gain quasiment linéaire.

5.4 Extension pour les problèmes multichamps

Dans cette section, on précise la définition de la PGD (ainsi que les algorithmes associés) dans le cas de problèmes multichamps. On rappelle que dans notre implémentation, le problème à F champs consiste à trouver le F-tuple de tenseurs $[\mathbf{u}]$ tel que

$$[\![\mathbf{A}]\!] \overset{\mathrm{D}}{\cdot} [\mathbf{u}] = [\mathbf{b}]\,, \tag{5.26}$$

où les composantes $\mathbf{A}_{ij}$ et $\mathbf{b}_i$ sont données pour $i, j = 1, \cdots, F$ sous format séparé par

$$\mathbf{A}_{ij} = \sum_{k=1}^{M_A(i,j)} \mathbf{A}^S_{ijk} \otimes \mathbf{A}^T_{ijk} \quad \text{et} \quad \mathbf{b}_i = \sum_{k=1}^{M_b(i)} \boldsymbol{b}^S_{ik} \otimes \boldsymbol{b}^T_{ik}\,. \tag{5.27}$$

Séparation de variables dans le cas multichamps

Dans le cadre de la PGD, la séparation de variables espace-temps a été utilisée pour traiter des problèmes multichamps par [Dureisseix *et al.*, 2003, Néron et Dureisseix, 2008, Néron, 2010, Beringhier *et al.*, 2010] [18]. Dans toutes ces contributions, chaque composante $\mathbf{u}_i$ du F-tuple $[\mathbf{u}]$ est approchée avec une décomposition différente, c'est-à-dire que les modes espace-temps sont différents

18. Une séparation en variables spatiales a également été appliquée pour approcher des champs de vecteurs par [Dumon *et al.*, 2011, Bognet *et al.*, 2012].

d'un champ à un autre. Une approximation $[\mathbf{u}]_M$ de $[\mathbf{u}]$ est donc cherchée dans le sous-ensemble $\mathrm{R}_{F,M}$ défini par

$$\mathrm{R}_{F,M} = \left\{ [\mathbf{u}] \mid \mathbf{u}_i \in \mathbb{R}^{n_S(i)} \otimes \mathbb{R}^{n_T(i)} \text{ et } \mathbf{u}_i = \sum_{m=1}^{M} \boldsymbol{w}_{im} \otimes \boldsymbol{\lambda}_{im} \text{ pour } i = 1, \cdots, F \right\}, \tag{5.28}$$

où on utilise le même rang M pour approcher chaque tenseur $\mathbf{u}_i$ [19].

Définition de la meilleure approximation dans la norme canonique

Afin de définir la meilleure approximation de rang M, on introduit la « norme canonique » (notée $\| \, \|_2$) dans le cas multichamps. Cette norme est définie (sur l'ensemble des F-tuples de tenseurs d'ordre $D = 2$) par

$$\| [\mathbf{u}] \|_2 = \left([\mathbf{u}] \overset{\mathrm{D}}{\cdot} [\mathbf{u}] \right)^{1/2} = \left(\sum_{i=1}^{F} \mathbf{u}_i \overset{\mathrm{D}}{\cdot} \mathbf{u}_i \right)^{1/2}. \tag{5.29}$$

On définit alors la meilleure approximation $[\mathbf{u}]_M$ de $[\mathbf{u}]$ dans $\mathrm{R}_{F,M}$ au sens de cette norme par

$$\boxed{[\mathbf{u}]_M = \arg \min_{[\mathbf{u}]^\star \in \mathrm{R}_{F,M}} \| [\mathbf{u}] - [\mathbf{u}]^\star \|_2}. \tag{5.30}$$

Remarque 5.13. *Avec le choix de la norme $\| \, \|_2$, le problème (5.30) revient à appliquer le problème (5.5) pour chaque composante $\mathbf{u}_i$, pour $i = 1, \cdots, F$. Dans la suite de cette section, on verra que la définition de la meilleure approximation est moins triviale dans le cas d'une norme énergétique.*

Remarque 5.14. *Les composantes du F-tuple $[\mathbf{u}]$ peuvent être associées à différentes physiques. La norme définie à l'équation (5.29) implique donc la sommation de quantités qui ne sont pas nécessairement homogènes en termes d'unités physiques. Aussi, un changement de variables doit être effectué afin de garantir l'homogénéité des différentes quantités sommées dans la norme multichamps* [20].

Procédures de partitionnement ou approches monolithiques

On distingue traditionnellement, deux grandes classes de méthodes pour résoudre des problèmes multi-champs, à savoir les procédures de partitionnement et les approches monolithiques. Dans les procédures de partitionnement, les différentes

19. Dans le cas où l'erreur due à une approximation de rang M est très différente pour chaque champ, il serait avantageux de considérer des rangs différents pour approcher chaque champ. Une telle approche nécessiterait cependant de « casser » la procédure monolitique décrite dans la suite de section. Elle demanderait également de pouvoir évaluer l'erreur de décomposition, à une itération donnée, pour chaque champ. On se contente ici d'utiliser le même rang pour chaque champ.

20. Le changement de variables utilisé pour les formulations en déplacement-vitesse utilisées dans ce manuscrit, sera précisé à la fin du chapitre.

	a priori $J([\mathbf{u}]^\star;[\![\mathbf{A}]\!],[\mathbf{b}])$	**a posteriori** $J([\mathbf{u}]^\star;[\mathbf{u}])$
(G)PGD	$\frac{1}{2}\langle[\mathbf{u}]^\star,[\mathbf{u}]^\star\rangle_{[\![\mathbf{A}]\!]}-\langle[\mathbf{u}]^\star,[\mathbf{b}]\rangle_2$	$\frac{1}{2}\left\|[\mathbf{u}]-[\mathbf{u}]^\star\right\|^2_{[\![\mathbf{A}]\!]}$
(R)PGD	$\frac{1}{2}\left\|[\mathbf{b}]-[\![\mathbf{A}]\!]\,\underline{\underline{\mathrm{D}}}\,[\mathbf{u}]^\star\right\|^2_2$	$\frac{1}{2}\left\|[\mathbf{u}]-[\mathbf{u}]^\star\right\|^2_{[\![\mathbf{A}]\!]'\,\underline{\underline{\mathrm{D}}}\,[\![\mathbf{A}]\!]}$
SVD	$\frac{1}{2}\left\|[\mathbf{b}]-[\![\mathbf{A}]\!]\,\underline{\underline{\mathrm{D}}}\,[\mathbf{u}]^\star\right\|^2_{([\![\mathbf{A}]\!]\,\underline{\underline{\mathrm{D}}}\,[\![\mathbf{A}]\!]')^{-1}}$	$\frac{1}{2}\left\|[\mathbf{u}]-[\mathbf{u}]^\star\right\|^2_2$
POD	$\frac{1}{2}\left\|[\mathbf{b}]-[\![\mathbf{A}]\!]\,\underline{\underline{\mathrm{D}}}\,[\mathbf{u}]^\star\right\|^2_{([\![\mathbf{A}]\!]\,\underline{\underline{\mathrm{D}}}\,[\![\mathbf{N}]\!]^{-1}\,\underline{\underline{\mathrm{D}}}\,[\![\mathbf{A}]\!]')^{-1}}$	$\frac{1}{2}\left\|[\mathbf{u}]-[\mathbf{u}]^\star\right\|^2_{[\![\mathbf{N}]\!]}$

TABLE 5.4: Définitions a priori et a posteriori de la fonctionnelle J dans le cas multichamps.

physiques du problème sont traitées séparément [Felippa *et al.*, 2001]. Ces approches ont notamment été couplées avec la PGD par [Dureisseix *et al.*, 2003, Néron, 2010]. Un des principaux avantages des procédures de partitionnement est la possibilité d'utiliser différentes discrétisations en espace et en temps pour chaque physique [Néron et Dureisseix, 2008]. La deuxième classe de méthode concerne les approches monolithiques. Ces approches reposent sur une formulation du problème sous la forme d'un unique système d'équations. Le problème (5.26) est par exemple formulé de façon monolithique (sur le domaine espace-temps). Les méthodes monolitiques sont généralement préférées pour leurs propriétés numériques (convergence, stabilité, précision), notamment lorsque les différentes physiques présentent un couplage fort [Michler *et al.*, 2004]. Une des principales limitations de ces méthodes est rencontrée lorsque le problème est formulé de façon monolitique en espace et que l'on souhaite utiliser une discrétisation en temps différente pour chaque physique. Cette difficulté est levée en écrivant le problème de façon monolitique sur le domaine espace-temps, comme c'est le cas dans ce manuscrit.

D'autre part, la définition d'un résidu multichamps n'est pas triviale dans le cas d'une procédure de partitionnement. Aussi, on privilégie ici une approche monolitique sur le domaine espace-temps, pour laquelle la définition du résidu multichamps est plus naturelle.

Définition de la meilleure approximation dans une norme énergétique

Toutes les définitions introduites dans le cas à un champ, peuvent être généralisées dans le cas multichamps sous la forme : trouver $[\mathbf{u}]_M \in \mathrm{R}_{F,M}$ tel que

$$\boxed{[\mathbf{u}]_M = \arg\min_{[\mathbf{u}]^\star\in\mathrm{R}_{F,M}} J([\mathbf{u}]^\star)}, \tag{5.31}$$

où la fonctionnelle J est donnée dans le cas multichamps dans le Tableau 5.4.

Le point clé concerne la procédure de minimisation alternée que l'on choisit pour résoudre ce problème. On adopte ici une procédure monolithique, alternativement en espace et en temps. C'est-à-dire que l'on minimise alternativement la fonctionnelle par rapport à tous les modes en espace et à tous les modes en temps. Dans le cas d'une approximation de rang un, cette procédure s'écrit de la façon suivante :

- connaissant $\begin{bmatrix} \boldsymbol{\lambda}_1 \\ | \\ \boldsymbol{\lambda}_F \end{bmatrix} \in \mathbb{R}^{n_T(1)+\cdots+n_T(F)}$, trouver $\begin{bmatrix} \boldsymbol{w}_1 \\ | \\ \boldsymbol{w}_F \end{bmatrix} \in \mathbb{R}^{n_S(1)+\cdots+n_S(F)}$ tels que :

$$\begin{bmatrix} \boldsymbol{w}_1 \\ | \\ \boldsymbol{w}_F \end{bmatrix} = \arg \min_{\begin{bmatrix} \boldsymbol{w}_1^\star \\ | \\ \boldsymbol{w}_F^\star \end{bmatrix} \in \mathbb{R}^{n_S(1)+\cdots+n_S(F)}} J(\begin{bmatrix} \boldsymbol{w}_1^\star \otimes \boldsymbol{\lambda}_1 \\ | \\ \boldsymbol{w}_F^\star \otimes \boldsymbol{\lambda}_F \end{bmatrix}), \tag{5.32a}$$

- connaissant $\begin{bmatrix} \boldsymbol{w}_1 \\ | \\ \boldsymbol{w}_F \end{bmatrix} \in \mathbb{R}^{n_S(1)+\cdots+n_S(F)}$, trouver $\begin{bmatrix} \boldsymbol{\lambda}_1 \\ | \\ \boldsymbol{\lambda}_1 \end{bmatrix} \in \mathbb{R}^{n_T(1)+\cdots+n_T(F)}$ tels que :

$$\begin{bmatrix} \boldsymbol{\lambda}_1 \\ | \\ \boldsymbol{\lambda}_F \end{bmatrix} = \arg \min_{\begin{bmatrix} \boldsymbol{\lambda}_1^\star \\ | \\ \boldsymbol{\lambda}_F^\star \end{bmatrix} \in \mathbb{R}^{n_T(1)+\cdots+n_T(F)}} J(\begin{bmatrix} \boldsymbol{w}_1 \otimes \boldsymbol{\lambda}_1^\star \\ | \\ \boldsymbol{w}_F \otimes \boldsymbol{\lambda}_F^\star \end{bmatrix}), \tag{5.32b}$$

La procédure est similaire dans le cas d'une approximation dans $\mathrm{R}_{F,M}$.

Remarque 5.15. *Les problèmes de minimisation (5.32) conduisent à la résolution alternative de systèmes linéaires de tailles respectives* $(\sum_{i=1}^{F} n_S(i)) \times (\sum_{i=1}^{F} n_S(i))$ *et* $(\sum_{i=1}^{F} n_T(i)) \times (\sum_{i=1}^{F} n_T(i))$. *On pourrait partitionner ces problèmes de façon à n'avoir à résoudre que des systèmes de tailles* $n_S(i) \times n_S(i)$ *ou* $n_T(i) \times n_T(i)$ *pour* $i = 1, \cdots, F$. *En pratique, on observe cependant qu'un tel partitionnement dégrade la vitesse de convergence du processus de minimisations alternées.*

Algorithmes dans le cas multichamps

Les prototypes des algorithmes utilisés pour calculer l'approximation de rang M sont très similaires au cas d'un problème à un champ. Aussi, ils ne sont pas détaillés dans le cas de problèmes multichamps. Les principales différences sont les suivantes :

- Les mappings sont écrits dans le cas multichamps. Pour la définition (G)PGD et la construction gloutonne, on a par exemple :

$$S_{\text{glouton}} : \begin{cases} \mathbb{R}^{n_T(1)+\cdots+n_T(F)} \longrightarrow \mathbb{R}^{n_S(1)+\cdots+n_S(F)} \\ \begin{bmatrix} \boldsymbol{\lambda}_1 \\ | \\ \boldsymbol{\lambda}_F \end{bmatrix} \longmapsto \begin{bmatrix} \boldsymbol{w}_1 \\ | \\ \boldsymbol{w}_F \end{bmatrix} \end{cases} \quad \text{tel que} \quad \begin{bmatrix} \mathbf{S}_{11} & - & \mathbf{S}_{1F} \\ | & \diagdown & | \\ \mathbf{S}_{F1} & - & \mathbf{S}_{FF} \end{bmatrix} . \begin{bmatrix} \boldsymbol{w}_1 \\ | \\ \boldsymbol{w}_F \end{bmatrix} = \begin{bmatrix} \boldsymbol{f}_1^S \\ | \\ \boldsymbol{f}_F^S \end{bmatrix}, \tag{5.33a}$$

$$T_{\text{glouton}} : \begin{cases} \mathbb{R}^{n_S(1)+\cdots+n_S(F)} \longrightarrow \mathbb{R}^{n_T(1)+\cdots+n_T(F)} \\ \begin{bmatrix} \boldsymbol{w}_1 \\ | \\ \boldsymbol{w}_F \end{bmatrix} \longmapsto \begin{bmatrix} \boldsymbol{\lambda}_1 \\ | \\ \boldsymbol{\lambda}_F \end{bmatrix} \end{cases} \quad \text{tel que} \quad \begin{bmatrix} \mathbf{T}_{11} & - & \mathbf{T}_{1F} \\ | & \diagdown & | \\ \mathbf{T}_{F1} & - & \mathbf{T}_{FF} \end{bmatrix} . \begin{bmatrix} \boldsymbol{\lambda}_1 \\ | \\ \boldsymbol{\lambda}_F \end{bmatrix} = \begin{bmatrix} \boldsymbol{f}_1^T \\ | \\ \boldsymbol{f}_F^T \end{bmatrix}, \tag{5.33b}$$

où les opérateurs $\mathbf{S}_{ij}$ et $\mathbf{T}_{ij}$ sont donnés pour $i, j = 1, \cdots, F$ par

$$\mathbf{S}_{ij} = \sum_{k=1}^{M_A(i,j)} (\boldsymbol{\lambda}_i.\mathbf{A}^T_{kij}.\boldsymbol{\lambda}_j)\mathbf{A}^S_{kij} \quad \text{et} \quad \mathbf{T}_{ij} = \sum_{k=1}^{M_A(i,j)} (\boldsymbol{w}_i.\mathbf{A}^S_{kij}.\boldsymbol{w}_j)\mathbf{A}^T_{kij}, \tag{5.33c}$$

et les vecteurs $\boldsymbol{f}^S_i$ et $\boldsymbol{f}^T_i$ sont donnés pour $i = 1, \cdots, F$ en fonction du résidu $[\mathbf{r}]$ (où $\mathbf{r}_i = \sum_{k=1}^{M_r(i)} \boldsymbol{r}^S_{ki} \otimes \boldsymbol{r}^T_{ki}$) par

$$\boldsymbol{f}^S_i = \sum_{k=1}^{M_r(i)} (\boldsymbol{\lambda}_i.\boldsymbol{r}^T_{ki})\boldsymbol{r}^S_{ki} \quad \text{et} \quad \boldsymbol{f}^T_i = \sum_{k=1}^{M_r(i)} (\boldsymbol{w}_i.\boldsymbol{r}^S_{ki})\boldsymbol{r}^T_{ki}. \tag{5.33d}$$

- Les opérateurs du problème symétrisé sont donnés par $[\![\mathbf{A}]\!]'^{\mathrm{D}}.[\![\mathbf{A}]\!]$ et $[\![\mathbf{A}]\!]'^{\mathrm{D}}.[\mathbf{b}]$.
- Pour la construction gloutonne avec projection, chaque composante $\mathbf{u}_i$ est approchée sous la forme de l'équation (5.22).
- La procédure de normalisation est appliquée sur chacune des composantes $\boldsymbol{w}_i$ et $\boldsymbol{\lambda}_i$ pour $i = 1, \cdots, F$.
- Le critère d'arrêt est appliqué sur la fonctionnelle du problème multichamps.

Remarque 5.16. *Dans le cas d'un problème multichamps, l'étude de la complexité des algorithmes est similaire au cas à un champ décrit dans l'Exemple 5.1. On notera cependant que la taille des mappings dépend du nombre de champs (voir la Remarque 5.15). Une autre différence concerne le rang des composantes de l'opérateur du problème symétrisé. Dans le cas d'un problème multichamps, on a* $M_{A'A}(i,j) = \sum_{k=1}^{F} M_A(k,i)M_A(k,j)$ *et l'assemblage des mappings peut donc rapidement devenir très coûteux.*

Remarque 5.17. *En pratique, tous les algorithmes ont été implémentés de façon générique (avec le logiciel Matlab) pour un problème à* F*-champs et l'approximation de tenseurs d'ordre* D *dans* $\mathrm{R}^D_{F,M}$.

5.5 Application à l'équation des ondes

Dans cette section, les différentes définitions de la PGD sont appliquées à l'équation des ondes. On présente tout d'abord les différents cas tests étudiés. Puis la convergence de l'algorithme glouton est analysée pour toutes les définitions de la PGD introduites précédemment. On compare ensuite les différents algorithmes pour la définition en minimum de résidu de la PGD. Quelques remarques sont émises concernant le cas des problèmes multichamps. Enfin, la précision de l'approximation de rang M obtenue avec la PGD est comparée avec la meilleure approximation de rang M au sens de la norme $\| \ \|_2$.

Description des cas tests

La PGD est ici appliquée à l'équation des ondes dans un milieu unidimensionnel de longueur $L = 1\text{m}$ et sur une durée de $T = 5\text{s}$. La célérité des ondes dans le milieu est fixée à 1m/s. De cette façon, une onde parcourt 5 fois la distance L. On considère les quatre configurations présentées sur la Figure 5.3. Chacune d'elles correspond à l'activation d'une condition aux limites ou initiale différente, à savoir :

- un effort ponctuel $p(t)$ imposé au point $x = 0$ de type choc d'une durée ΔT telle que $\kappa = 10$ (on rappelle que $\Delta T = \frac{L}{c\sqrt{\kappa}}$),
- un déplacement $g(t)$ imposé au point $x = 0$ de type choc d'une durée telle que $\kappa = 10$,
- un déplacement initial $u_0(x)$ de type compression (on prend $u_0(x) = \frac{x}{L} - 1$),
- une vitesse initiale $v_0(x)$ constante (on prend $v_0(x) = v_0$ pour $x \in [0, L[$ et $v_0(L) = 0 \neq v_0$).

Pour tous les cas tests, une condition de Dirichlet est imposée nulle au point $x = L$.

Le problème espace-temps est discrétisé avec les méthodes d'approximation présentées au Chapitre 1. Sauf mention contraire, on utilise des éléments finis P1 en espace et les méthodes d'approximation en temps suivantes : le schéma de Newmark (avec $\beta = 1/4, \gamma = 1/2$), les méthodes de Galerkin discontinues en temps à un champ (TDG P2) et deux champs (TDG P1-P1), et la méthode de Galerkin continue en temps à deux champs (TG P1-P1). Pour tous les cas tests, on utilise les mêmes paramètres du maillage espace-temps, à savoir $N_S = 50$ éléments en espace et $N_T = 100$ intervalles de temps (sauf mention contraire).

La solution du problème espace-temps obtenue pour les différents cas tests et les différentes méthodes d'approximation est représentée sur la Figure 5.2 (on a représenté seulement le champ de déplacement).

On conserve le point de vue discret dans les notations. On rappelle que $\mathbf{U}$ est la représentation discrète du champ de déplacement, et $\mathbf{V}$ la représentation discrète du champ de vitesse. On note $\mathbf{U}_M$ (resp. $\mathbf{V}_M$) l'approximation de rang M de $\mathbf{U}$ (resp. $\mathbf{V}$).

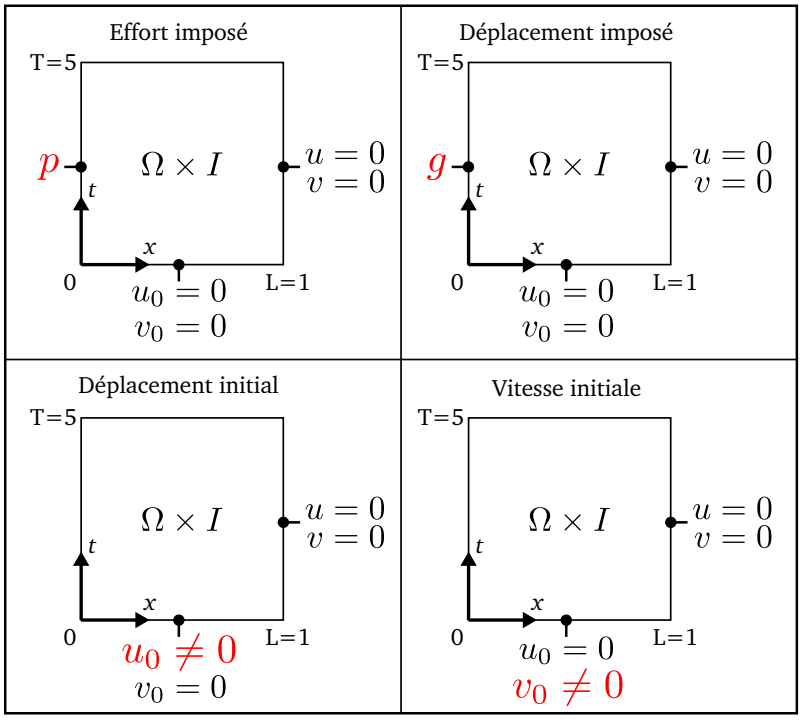

FIGURE 5.1: Conditions aux limites et initiales pour les différents cas tests.

Remarque 5.18. *Dans le cas d'un problème multichamps faisant intervenir des quantités physiques différentes (ici le déplacement et la vitesse), la norme associée à l'opérateur du problème espace-temps engendre la sommation de quantités qui ne sont pas homogènes en termes d'unités (voir la Remarque 5.14). Aussi, on remplace ici le champ de vitesse* $\mathbf{V}$ *par* $\tilde{\mathbf{V}} = \theta\mathbf{V}$ *et on procède de même pour le champ test* $\mathbf{V}^\star$*. Cette stratégie revient à remplacer les opérateurs* $[\![\mathbf{A}]\!]$ *et* $[\mathbf{b}]$ *du problème multichamps par* $[\![\mathbf{P}]\!]^D_\cdot[\![\mathbf{A}]\!]^D_\cdot[\![\mathbf{P}]\!]$ *et* $[\![\mathbf{P}]\!]^D_\cdot[\mathbf{b}]$ *avec*

$$[\![\mathbf{P}]\!] = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \frac{1}{\theta}\mathbf{I} \end{bmatrix}, \tag{5.34}$$

où $\mathbf{I}$ *et* $\mathbf{0}$ *sont les tenseurs identités et zéros respectivement. Différents choix sont alors possibles pour le paramètre* θ*. On prend ici* $\theta = \Delta t$*. On trouvera des détails supplémentaires dans [Boucinha* et al.*, 2013b] concernant les unités physiques des modes spatiaux et temporels. Dans la suite de cette section, on note indifféremment* $\mathbf{V}$ *et* $\tilde{\mathbf{V}}$*.*

5.5.1 Comparaison des définitions

On compare ici les différentes définitions de la PGD introduites dans ce chapitre, à savoir :

- la décomposition avec critère d'orthogonalité de Galerkin (G)PGD,
- la décomposition avec critère d'orthogonalité de Petrov-Galerkin (PG)PGD,

- la décomposition avec critère d'orthogonalité de Petrov-Galerkin, appliquée sur le problème symétrisé (PGsym)PGD,
- la décomposition en minimum de résidu dans la norme canonique (R)PGD,
- la décomposition en minimum de résidu dans une norme idéale (IMR)PGD (voir le chapitre suivant) [21].

On utilise l'algorithme de construction gloutonne pure pour calculer l'approximation de rang M associée à chacune de ses définitions. Les paramètres de l'algorithme sont fixés à $\xi_{\max} = 50$ et $\epsilon_{\max} = 10^{-3}$. Pour chacune des définitions, on évalue l'erreur relative entre la solution discrète en déplacement ($\mathbf{U}$) et son approximation de rang M ($\mathbf{U}_M$) définie par

$$\text{err}(\mathbf{U}_M) = \frac{\left\| \mathbf{U} - \mathbf{U}_M \right\|_2}{\left\| \mathbf{U} \right\|_2}. \tag{5.35}$$

La construction est stoppée à un rang M tel que :

- $\text{err}(\mathbf{U}_M) > 10^1$ (l'algorithme diverge),
- $\text{err}(\mathbf{U}_M) \leq 2.10^{-7}$ (l'algorithme a totalement convergé),
- $2.10^{-7} < \text{err}(\mathbf{U}_M) < 10^1$ et $M \geq 500$ (l'algorithme converge très lentement).

Afin de comparer la robustesse des différentes définitions de la PGD, les calculs sont réalisés pour tous les cas tests et toutes les méthodes d'approximation décrites précédemment. Les résultats obtenus sont présentés sur la Figure 5.2 où l'on a représenté l'erreur $\text{err}(\mathbf{U}_M)$ en fonction du rang M. Les courbes noires représentent les approximations PGD et les courbes rouges la meilleure approximation de rang M.

On peut faire les commentaires suivants :

- La définition (G)PGD diverge dans tous les cas testés.
- La définition (PG)PGD converge si l'équation des ondes est discrétisée avec la méthode TG P1-P1. On observe cependant de fortes oscillations de l'erreur en fonction du rang M. Cette définition diverge dans tous les autres cas testés.
- La définition (PGsym)PGD converge dans tous les cas testés. Pour la majorité des cas, il faut un rang $M > 500$ pour que l'erreur soit inférieure à 2.10^{-7}. On observe également de fortes oscillations de l'erreur en fonction du rang M.
- La définition (R)PGD converge dans tous les cas testés. L'erreur 2.10^{-7} est obtenue pour presque tous les cas avec un rang $M < 500$. La convergence de l'erreur en fonction du rang M est marquée par une succession de phases de stagnation puis convergence très rapide [22].

21. Pour cette définition, on utilise une construction gloutonne avec un seul mode auxiliaire.
22. Grossièrement, on observe « des marches » sur les courbes de convergence.

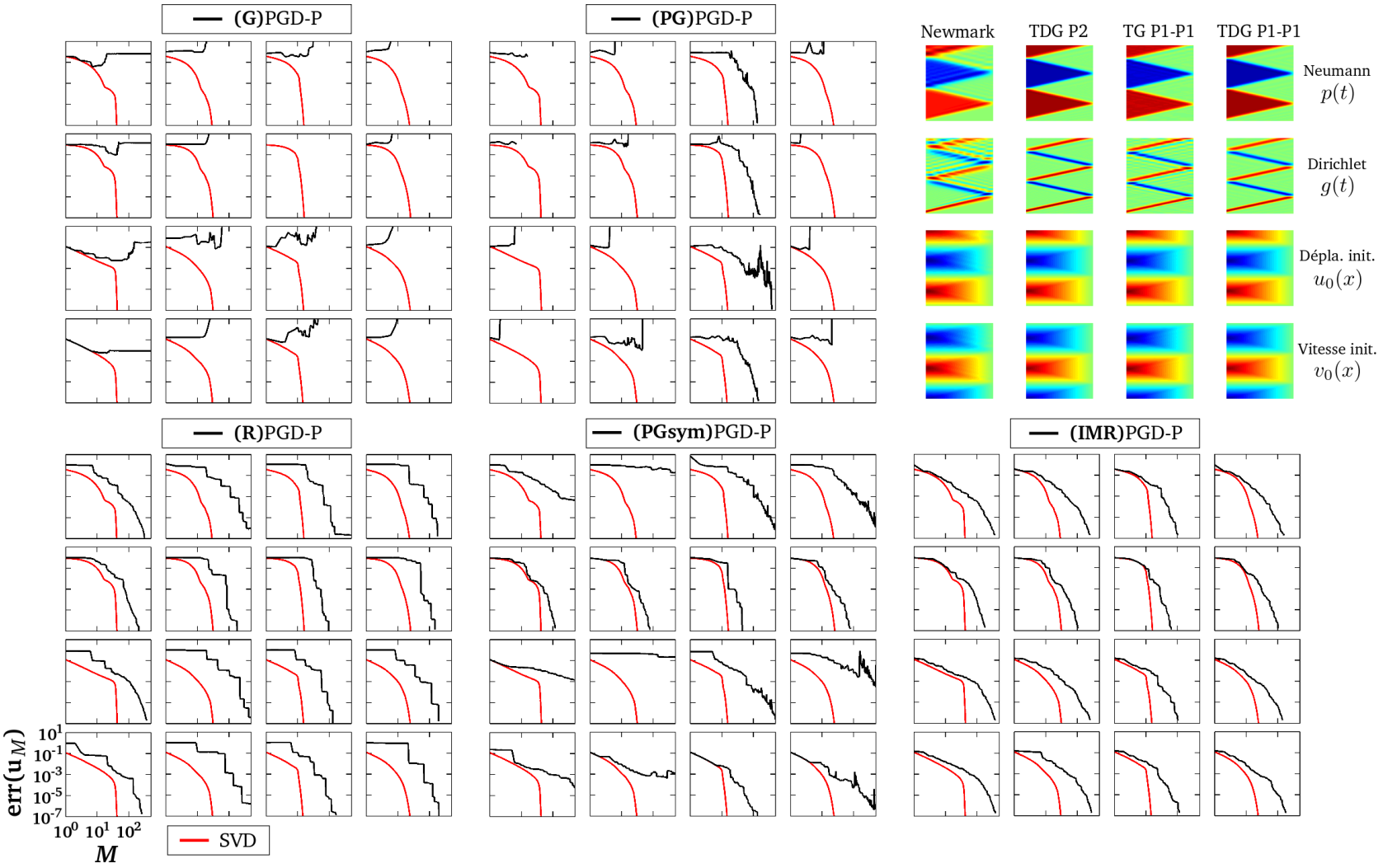

FIGURE 5.2: Erreur relative entre la solution (en déplacement) du problème espace-temps et son approximation de rang M.

- La définition (IMR)PGD converge dans tous les cas testés. L'erreur 2.10^{-7} est obtenue dans tous les cas avec un rang $M < 500$. Pour les faibles valeurs de M (typiquement $M \leq 10$), l'erreur obtenue avec cette définition de la PGD est très proche de l'erreur obtenue avec la SVD.

Pour les problèmes de dynamique transitoire testés, les définitions de la PGD les plus robustes sont donc celles en minimum de résidu [23]. Dans la suite de ce chapitre, on analyse plus en détails la définition en minimum de résidu dans la norme canonique (R)PGD. La description de la définition en minimum de résidu dans une norme idéale est l'objet du chapitre suivant.

5.5.2 Comparaison des algorithmes

On compare ici les différents algorithmes présentés dans ce chapitre pour calculer la PGD. On considère uniquement la définition en minimum de résidu (R)PGD. On compare les constructions gloutonne pure (PGD-P), gloutonne avec projection (PGD-P+proj), gloutonne avec mise à jour (PGD-P+upd avec $\xi_{\text{upd}} = 1$ ou 5), et directe (PGD-S). Les calculs sont réalisés pour tous les cas tests et toutes les méthodes d'approximation. La construction est stoppée lorsque le résidu relatif est inférieur à 2.10^{-7} ou que le rang $M \geq 50$. Pour l'algorithme PGD-S, on calcule la décomposition jusqu'à $M = 20$. La convergence des algorithmes est analysée en terme du résidu relatif ou de l'erreur relative.

Convergence du résidu

Dans le cas d'un problème à un champ (déplacement), le résidu relatif est donné en fonction du rang M par

$$\text{res}(\mathbf{U}_M) = \frac{\left\| \mathbf{b} - \mathbf{A} \overset{\text{D}}{\cdot} \mathbf{U}_M \right\|_2}{\left\| \mathbf{b} \right\|_2}. \tag{5.36}$$

Dans le cas d'un problème à deux champs (déplacement-vitesse), on note $[\mathbf{u}] = [\mathbf{U}, \mathbf{V}]$ et $[\mathbf{u}]_M = [\mathbf{U}_M, \mathbf{V}_M]$, et le résidu relatif est donné en fonction du rang M par

$$\text{res}(\mathbf{U}_M, \mathbf{V}_M) = \frac{\left\| [\mathbf{b}] - [\![\mathbf{A}]\!] \overset{\text{D}}{\cdot} [\mathbf{u}]_M \right\|_2}{\left\| [\mathbf{b}] \right\|_2}. \tag{5.37}$$

Les résultats obtenus pour tous les algorithmes et les différents cas tests sont présentés sur la Figure 5.3 dans le cas d'un problème à un champ et sur la Figure 5.4 dans le cas d'un problème à deux champs. On se concentre ici sur l'analyse de la convergence en terme de résidu relatif en fonction du rang M.

On peut faire les commentaires suivants :

23. Cette conclusion semble naturelle dans la mesure où ce type de problème est non-symétrique. On rappelle que la convergence de la PGD n'a été démontrée pour des problèmes non-symétriques que pour la définition en minimum de résidu.

- Pour tous les algorithmes (R)PGD, le résidu décroît de façon monotomne lorsque la rang M augmente.
- L'algorithme (R)PGD-S donne la meilleure approximation de rang M au sens de la minimisation du résidu (on pourra notamment comparer le résidu à un rang donné, obtenu avec l'algorithme PGD-S ou avec la SVD).
- À un rang donné, l'algorithme (R)PGD-P est le moins précis.
- À un rang donné, la projection sur une base réduite (algorithme (R)PGD+proj) améliore légèrement le résidu par rapport à l'algorithme de construction gloutonne pure.
- La mise à jour de la construction gloutonne (algorithme (R)PGD+upd) permet d'obtenir une très bonne approximation de la meilleure approximation de rang M (au sens du minimum de résidu) donnée par l'algorithme (R)PGD-S. L'augmentation du nombre d'itérations de mise à jour (ξ_{upd}) améliore légèrement la précision de cette approximation.

Convergence de l'erreur

On analyse maintenant la convergence de l'approximation de rang M en terme d'erreur relative. L'évolution de l'erreur relative est représentée sur la Figure 5.3 pour le champ de déplacement (err($\mathbf{U}_M$)) et sur la Figure 5.4 pour les champs de déplacement et de vitesse (err($\mathbf{U}_M$) et err($\mathbf{V}_M$)). On peut faire les commentaires suivants :

- L'algorithme (R)PGD-S donne une bonne approximation de la SVD (qui est la meilleure approximation de rang M au sens de la minimisation de l'erreur dans la norme $\|\ \|_2$). Notamment, la minimisation du résidu multichamps permet d'obtenir une bonne approximation de la SVD du champ de déplacement et de vitesse.
- Globalement, l'analyse de la convergence de l'erreur met en évidence des phases de stagnation-convergence rapide, que l'on ne retrouve pas dans les courbes de convergence du résidu. Cette observation traduit le fait que la minimisation du résidu engendre des modes qui sont peu contribuants en terme de minimisation de l'erreur. On doit donc calculer de nombreux modes intermédiaires avant d'obtenir un mode contribuant au sens de la minimisation de l'erreur.

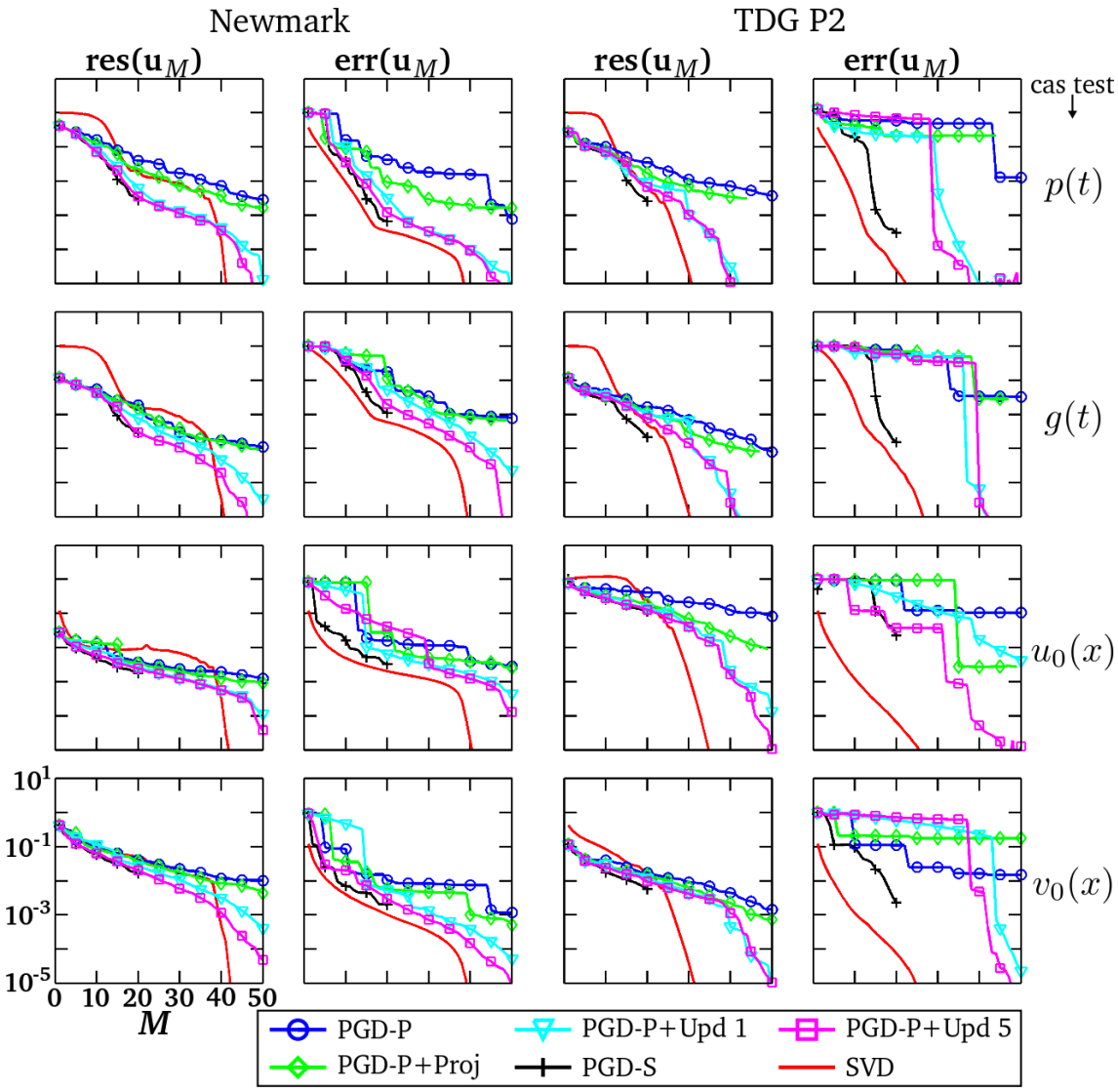

FIGURE 5.3: Erreur et résidu relatifs pour une approximation de rang M dans le cas de problèmes à un champs et de la définition (R)PGD.

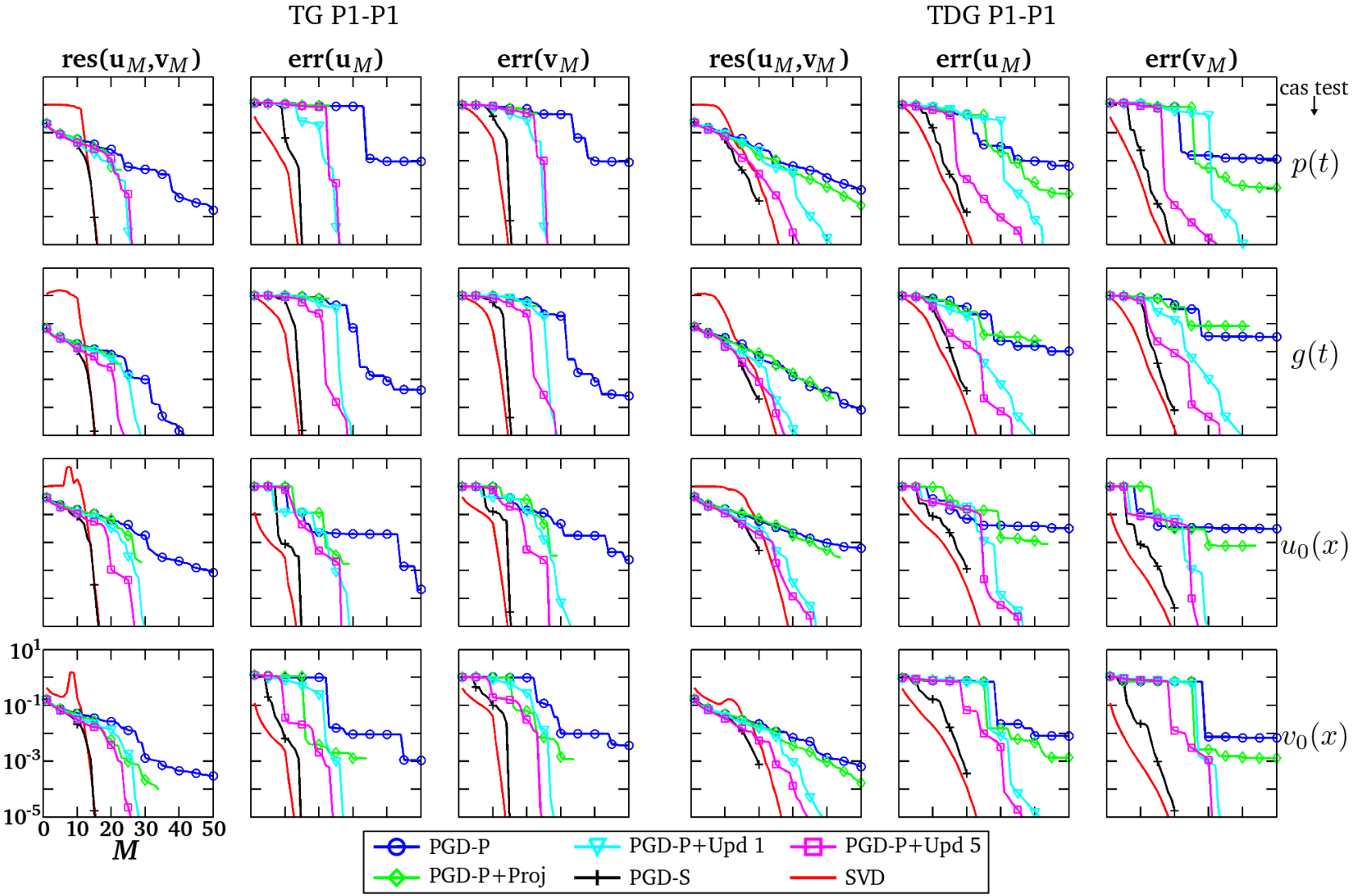

FIGURE 5.4: Erreur et résidu relatifs pour une approximation de rang M dans le cas de problèmes à deux champs et de la définition (R)PGD.

Influence des paramètres $\xi_{\max}$ et $\epsilon_{\max}$

Les paramètres $\xi_{\max}$ et $\epsilon_{\max}$, qui contrôlent la convergence du processus de minimisations alternées, jouent un rôle clé concernant la complexité des algorithmes PGD. Pour tous les cas testés, l'approximation de rang M obtenue avec $\xi_{\max} = 10$ itérations est quasiment identique à celle obtenue avec $\xi_{\max} = \infty$ et $\epsilon_{\max} = 10^{-6}$. Une étude détaillée a été effectuée dans [Boucinha *et al.*, 2013b] sur cet aspect et n'est pas représentée ici.

Efficacité par rapport à une résolution incrémentale

L'efficacité de la PGD en terme de temps de calcul, doit être comparée à la méthode de référence utilisée pour résoudre le problème espace-temps. On doit donc comparer la complexité des algorithmes PGD à la complexité d'une résolution incrémentale. On réalise ici une telle comparaison pour le cas test avec condition de Dirichlet et le schéma TDG P1-P1. On fixe tout d'abord les paramètres du maillage espace-temps de façon à ce que l'erreur de discrétisation soit égale à 1%. Pour le cas test choisi et le schéma TDG P1-P1, il faut $N_S = 224$ éléments pour le maillage spatial et $N_T = 1024$ intervalles de temps. Puis, on cherche le rang $M_{\min}$ tel que l'erreur due à une approximation de ce rang soit inférieure ou égale à 1%. En utilisant l'algorithme (R)PGD-P avec $\xi_{\max} = 10$, on obtient $M_{\min} = 293$. En mesurant le temps cpu avec le logiciel Matlab, il faut alors 6s pour obtenir la solution avec le schéma incrémental alors que la résolution de tous les mappings demande 39s et leur assemblage prend 75s. Pour ce cas test, le solveur (R)PGD-P est donc bien plus coûteux que le solveur incrémental.

La même étude avec l'algorithme (R)PGD-P+upd1 aboutit à $M_{\min} = 115$. Cependant, la résolution de tous les mappings (y compris ceux de l'étape de mise à jour) demande 112s et leur assemblage 107s. Une nouvelle fois, cet algorithme n'est pas compétitif par rapport à une résolution incrémentale.

5.5.3 Optimalité de l'approximation PGD

L'algorithme (R)PGD-S donne la meilleure approximation de rang M au sens de la minimisation de l'erreur dans la norme $\| \|_{\mathbf{A}'\mathrm{D}\mathbf{A}}$. Cependant, l'objectif poursuivit dans ce manuscrit est de calculer une bonne approximation de la meilleure approximation de rang M au sens de la minimisation de l'erreur dans la norme $\| \|_2$ (qui est la SVD de $\mathbf{U}$ tronquée au rang M). La comparaison effectuée sur les Figures 5.3 et 5.4 montre que la décomposition calculée avec l'algorithme (R)PGD-S est une relativement bonne estimation de la SVD. Néanmoins, le maillage espace-temps utilisé pour tracer ces courbes est très grossier. Et, lorsque l'on raffine celui-ci, on observe, pour une valeur du rang M fixé, que l'approximation calculée avec l'algorithme (R)PGD-S est de moins en moins précise en comparaison de la SVD tronquée au rang M (voir les résultats du Tableau 5.5).

	$\|\mathbf{U}-\mathbf{U}_{10}\|_2/\|\mathbf{U}\|_2$	
$N_S \times N_T$	SVD	(R)PGD-S
50×100	2e-02	6e-01
100×200	4e-02	9e-01
200×400	6e-02	9e-01
400×800	6e-02	1e+00

TABLE 5.5: Erreur relative due à une approximation de rang $M = 10$ en fonction des paramètres du maillage espace-temps (cas test avec condition de Dirichlet et discrétisation en temps avec TDG P1-P1, N_S et N_T sont le nombre d'éléments dans le maillage spatial et temporel respectivement).

Dès lors que la dimension de l'espace de discrétisation est grande, la définition classique de la PGD en minimum de résidu échoue donc à fournir une bonne approximation de la meilleure approximation de rang M au sens de la norme $\|\ \|_2$. Quel que soit l'algorithme utilisé, l'approximation obtenue avec cette définition de la PGD ne sera pas optimale au sens de la norme $\|\ \|_2$. De plus, pour un rang M donné, l'approximation (R)PGD sera d'autant moins précise que la dimension de l'espace de discrétisation sera grande.

Un moyen de résoudre ce problème d'optimalité, consiste à utiliser un algorithme de type (R)PGD pour construire une approximation non-optimale mais suffisamment précise de la solution, puis à calculer a posteriori une SVD de l'approximation PGD obtenue. On peut ensuite tronquer cette SVD de façon à supprimer les modes non contribuants au sens de la norme $\|\ \|_2$ (voir la Figure 5.5). Cette stratégie n'est cependant pas une méthode de réduction de modèle a priori, puisque l'on doit, dans une première étape, calculer une bonne approximation de la solution de référence (sous la forme d'une PGD contenant un nombre inutilement grand de modes espace-temps), avant de calculer la meilleure approximation de rang M de cette approximation au sens de la norme $\|\ \|_2$. Dans le cas d'un problème de dynamique transitoire, le coût associé au calcul d'une bonne approximation de la solution avec un algorithme de type (R)PGD est prohibitif (dans le cas représenté sur la Figure 5.5 plus de 75% des modes espace-temps sont inutiles).

5.6 Conclusion

Dans ce chapitre, un état de l'art des méthodes de réduction modèle *a priori*, basées sur la décomposition généralisée propre (PGD) a été proposé. Les définitions classiques de la PGD, ainsi que les algorithmes de construction associés, ont été détaillés dans un cadre générique à un champ puis étendus dans un cadre multichamps.

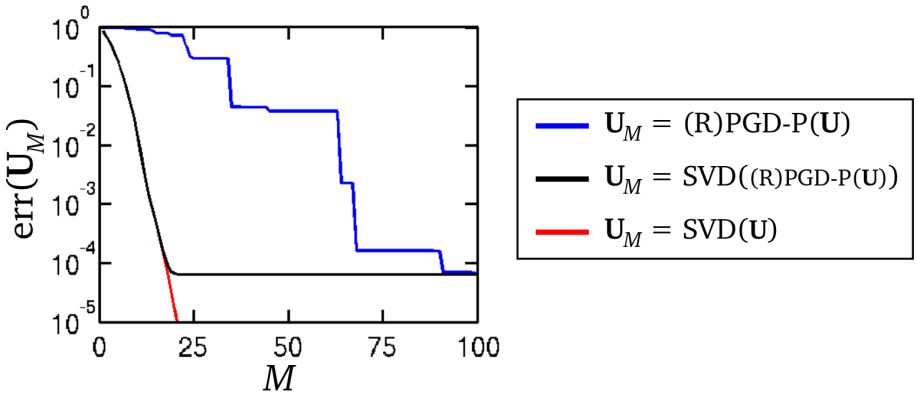

FIGURE 5.5: Illustration d'une procédure permettant d'obtenir une bonne approximation de la SVD de $\mathbf{U}$ à partir d'une approximation PGD non-optimale mais suffisamment précise.

Toutes les méthodes présentées ont été comparées sur quatres cas tests académiques de propagation d'ondes. Chaque cas test a été choisi de façon à activer une condition aux limites ou initiale différente (conditions de Dirichlet et de Neumann, conditions initiales en déplacement et en vitesse) permettant ainsi de valider la stratégie utilisée pour imposer ces conditions dans le cadre de la construction d'une approximation à variables séparées espace-temps.

Les résultats obtenus ont montré que l'approximation PGD minimisant la norme canonique du résidu est la plus robuste, pour approcher la solution d'un problème de dynamique transitoire. Cependant, quel que soit l'algorithme utilisé, cette approximation n'est pas optimale, ni même précise, en comparaison de l'approximation de rang M qui minimise l'erreur dans la norme $\| \ \|_2$. Aussi, cette définition classique de la PGD échoue à approcher efficacement la solution d'un problème de dynamique transitoire sous la forme d'une représentation à variables séparées espace-temps. On doit calculer un (trop) grand nombre de modes espace-temps pour obtenir une approximation suffisamment précise.

Cet échec est clairement lié à la dégradation de la précision de l'approximation (R)PGD, pour un rang M donné, lorsque la dimension de l'espace d'approximation augmente. On peut relier cette dégradation de la précision au mauvais conditionnement de l'opérateur du problème espace-temps. Récemment, certains auteurs ont proposé des stratégies de préconditionnement du système linéaire afin d'améliorer la précision de l'approximation PGD [24]. Dans le chapitre suivant, une nouvelle approche, introduite par [Billaud-Friess *et al.*, 2013] est présentée.

24. On pourra notamment consulter l'approche simplifiée proposée par [Barbarulo, 2012], ou les préconditionneurs plus génériques développés par [Giraldi *et al.*, 2013].

Chapitre 6

PGD par minimisation du résidu dans une norme idéale

Dans ce chapitre, une nouvelle définition de la PGD, récemment introduite par [Billaud-Friess et al.*, 2013], est appliquée à un problème académique d'élastodynamique transitoire. Cette nouvelle définition permet d'obtenir, a priori, une très bonne approximation de la meilleure approximation de rang M introduite au Chapitre 2.*

Sommaire

6.1 Introduction

La définition classique de la PGD, par minimisation du résidu dans la norme canonique, permet d'obtenir a priori une approximation à variables séparées espace-temps de la solution du problème de dynamique transitoire. Cependant, l'approximation obtenue de cette façon est très peu précise en comparaison de la meilleure approximation de rang M définie au Chapitre 2. Pour obtenir une approximation suffisamment précise de la solution de référence, un trop grand nombre de modes espace-temps doivent être calculés et stockés pour que la PGD classique soit efficace par rapport à un solveur incrémental. Dans ce chapitre, on présente une nouvelle stratégie, introduite par [Billaud-Friess *et al.*, 2013], qui permet d'obtenir, a priori, une très bonne approximation de la meilleure approximation de rang M définie au Chapitre 2, sans avoir à calculer un grande nombre de modes. Cette stratégie est ici appliquée, dans le cadre d'une formulation multi-champs, à la résolution d'un problème académique de dynamique transitoire dans un milieu bidimentionnel [Boucinha *et al.*, 2013a].

Définir *a priori* la meilleure approximation de rang M

On rappelle que la meilleure approximation de rang M de $\mathbf{u} \in \mathbb{R}^{n_S} \otimes \mathbb{R}^{n_T}$ consiste à trouver $\mathbf{u}_M \in \mathrm{R}_M$ tel que

$$\boxed{\mathbf{u}_M = \arg \min_{\mathbf{u}^\star \in \mathrm{R}_M} \left\| \mathbf{u} - \mathbf{u}^\star \right\|_2}, \tag{6.1}$$

où R_M est le sous-ensemble des décompositions espace-temps de rang M et $\left\| \ \right\|_2$ est la norme canonique sur $\mathbb{R}^{n_S} \otimes \mathbb{R}^{n_T}$. Aussi, l'objectif poursuivi dans ce manuscrit est de trouver cette meilleure approximation, sans connaitre le tenseur $\mathbf{u}$. Les seules données dont on dispose sur le tenseur $\mathbf{u}$, sont les opérateurs $\mathbf{A}$ et $\mathbf{b}$ du système linéaire dont il est solution (qui sont donnés sous format tensoriel). L'opérateur $\mathbf{A}$ étant non-symmétrique, on cherche, pour atteindre cet objectif, à définir un problème de minimisation du résidu dans une certaine norme.

L'idée générale est d'introduire une norme, telle que la minimisation du résidu dans cette norme, soit équivalente à minimiser l'erreur dans la norme $\left\| \ \right\|_2$. On cherche donc un opérateur symétrique défini positif $\mathbf{N}$ (définissant la « norme idéale » $\left\| \mathbf{x} \right\|_{\mathbf{N}} = \sqrt{\mathbf{x} \cdot^{\mathrm{D}} \mathbf{N} \cdot^{\mathrm{D}} \mathbf{x}}$) tel que

$$\left\| \mathbf{u} - \mathbf{u}^\star \right\|_2 = \left\| \mathbf{b} - \mathbf{A} \cdot^{\mathrm{D}} \mathbf{u}^\star \right\|_{\mathbf{N}} \tag{6.2}$$

Clairement, le choix $\mathbf{N} = (\mathbf{A} \cdot^{\mathrm{D}} \mathbf{A}')^{-1}$ vérifie ces conditions. Le problème (6.1) est donc équivalent à trouver $\mathbf{u}_M \in \mathrm{R}_M$ tel que

$$\boxed{\mathbf{u}_M = \arg \min_{\mathbf{u}^\star \in \mathrm{R}_M} \left\| \mathbf{b} - \mathbf{A} \cdot^{\mathrm{D}} \mathbf{u}^\star \right\|_{(\mathbf{A} \cdot^{\mathrm{D}} \mathbf{A}')^{-1}}}. \tag{6.3}$$

Le problème de minimisation (6.3) peut être interprété comme la définition *a priori* de la décomposition en valeur singulière (SVD) du tenseur $\mathbf{u}$, dans le cas où ce tenseur est solution d'un problème non-symétrique.

6.2 Description de l'algorithme

Bien sûr, le calcul de l'opérateur $(\mathbf{A} \overset{\mathrm{D}}{\cdot} \mathbf{A}')^{-1}$ est au moins aussi coûteux qu'une résolution directe du système linéaire. Aussi, l'objectif est maintenant, de trouver une bonne approximation de la solution du problème (6.3) sans avoir à calculer explicitement cet opérateur. L'algorithme introduit par [Billaud-Friess *et al.*, 2013] et présenté dans cette section, permet d'obtenir une telle approximation.

6.2.1 Construction directe

Pour résoudre la problème (6.3), [Billaud-Friess *et al.*, 2013] introduisent tout d'abord un algorithme de type gradient, qui peut être vue comme une extension au problème d'approximation non-linéaire dans $\mathbb{R}_M$, des approches proposées par [Cohen *et al.*, 2012, Dahmen *et al.*, 2012].

Algorithme de type gradient

À une itération ξ donnée, on suppose que l'on connait une approximation $\mathbf{u}_M^{(\xi)} \in \mathrm{R}_M$ de $\mathbf{u}_M$. Puis, pour éviter d'avoir à calculer explicitement l'opérateur de la norme idéale, on introduit une variable auxiliaire $\mathbf{y}^{(\xi)} \in \mathbb{R}^{n_S} \otimes \mathbb{R}^{n_T}$, telle que

$$\mathbf{y}^{(\xi)} = (\mathbf{A} \overset{\mathrm{D}}{\cdot} \mathbf{A}')^{-1} \overset{\mathrm{D}}{\cdot} (\mathbf{b} - \mathbf{A} \overset{\mathrm{D}}{\cdot} \mathbf{u}_M^{(\xi)}) . \tag{6.4a}$$

On calcule alors l'itéré suivant $\mathbf{u}_M^{(\xi+1)} \in \mathrm{R}_M$ en résolvant

$$\mathbf{u}_M^{(\xi+1)} = \arg \min_{\mathbf{u}^\star \in \mathrm{R}_M} \left\| \mathbf{u}_M^{(\xi)} + \mathbf{A}' \overset{\mathrm{D}}{\cdot} \mathbf{y}^{(\xi)} - \mathbf{u}^\star \right\|_2 . \tag{6.4b}$$

On peut vérifier que si l'équation (6.4a) est résolu exactement, alors cette algorithme converge vers la meilleure approximation de rang M en une itération, quelle que soit l'initialisation $\mathbf{u}_M^{(0)}$. Cependant, la résolution exacte de l'équation (6.4a) demande autant d'efforts que la résolution du système linéaire $\mathbf{A} \overset{\mathrm{D}}{\cdot} \mathbf{u} = \mathbf{b}$, et ne peut donc pas être réalisée en pratique. Aussi, l'idée proposée par [Billaud-Friess *et al.*, 2013] est d'introduire une perturbation de cette approche idéale.

Perturbation de l'algorithme de type gradient

Dans l'approche perturbée, l'étape (6.4a) est remplacée par une approximation de faible rang (noté $\mathbf{y}_R^{(\xi)} \in \mathrm{R}_R$) de $(\mathbf{A} \overset{\mathrm{D}}{\cdot} \mathbf{A}')^{-1} \overset{\mathrm{D}}{\cdot} (\mathbf{b} - \mathbf{A} \overset{\mathrm{D}}{\cdot} \mathbf{u}_M^{(\xi)})$. Différentes stratégies peuvent alors être employées pour calculer cette approximation. Le problème auxiliaire étant symétrique, on choisit ici de calculer celle-ci avec la PGD de type Galerkin. Pour réduire les coûts de calcul, on utilise une construction gloutonne. L'approche perturbée obtenue de cette façon est appelée (IMR)PGD (« Ideal Minimal Residual PGD »). Elle

s'écrit : connaissant $\mathbf{u}_M^{(\xi)} \in \mathrm{R}_M$, trouver $\mathbf{u}_M^{(\xi+1)} \in \mathrm{R}_M$ tel que

$$\mathbf{r}^{(\xi)} = \mathbf{b} - \mathbf{A} \overset{\mathrm{D}}{\cdot} \mathbf{u}_M^{(\xi)}, \tag{6.5a}$$

$$\mathbf{y}_R^{(\xi)} = \textbf{(G)PGD-P}(\mathbf{A} \overset{\mathrm{D}}{\cdot} \mathbf{A}', \mathbf{r}^{(\xi)}, R), \tag{6.5b}$$

$$\mathbf{u}_M^{(\xi+1)} = \arg \min_{\mathbf{u}^\star \in \mathrm{R}_M} \left\| \mathbf{u}_M^{(\xi)} + \mathbf{A}' \overset{\mathrm{D}}{\cdot} \mathbf{y}_R^{(\xi)} - \mathbf{u}^\star \right\|_2, \tag{6.5c}$$

où l'opérateur **(G)PGD-P**$(\mathbf{A}, \mathbf{b}, M)$ donne une approximation de rang M de $\mathbf{A}^{-1} \overset{\mathrm{D}}{\cdot} \mathbf{b}$, calculée avec l'algorithme glouton de la PGD de type Galerkin.

Convergence de l'algorithme

Dans l'algorithme proposé par [Billaud-Friess *et al.*, 2013], le rang R est choisi adaptativement (à chaque itération ξ) de façon à respecter une certaine précision. De cette manière, les auteurs ont démontré la converge de $\mathbf{u}_M^{(\xi)}$ vers un voisinage de $\mathbf{u}_M$, dont la précision peut être contrôlée en adaptant le rang R. Cependant, le critère permettant d'arrêter cette procédure engendre un coût supplémentaire non-négligeable [1]. Aussi, dans la stratégie présentée ici, le rang R est pris constant pour toutes les itérations et la convergence de l'algorithme est vérifiée numériquement.

Estimateur d'erreur

Un autre aspect important de la stratégie proposée par [Billaud-Friess *et al.*, 2013], est de pouvoir estimer l'erreur $\left\| \mathbf{u} - \mathbf{u}_M^{(\xi)} \right\|_2$ par une simple évaluation de $\left\| \mathbf{y}_R^{(\xi)} \right\|_{\mathbf{A} \overset{\mathrm{D}}{\cdot} \mathbf{A}'}$. Ils montrent en effet, sous les hypothèses suivantes,

1. à chaque itération (ξ), l'approximation $\mathbf{y}_R^{(\xi)}$ de $\mathbf{y}^{(\xi)}$ vérifie

$$\left\| \mathbf{y}^{(\xi)} - \mathbf{y}_R^{(\xi)} \right\|_{\mathbf{A} \overset{\mathrm{D}}{\cdot} \mathbf{A}'} \leq \delta \left\| \mathbf{y}^{(\xi)} \right\|_{\mathbf{A} \overset{\mathrm{D}}{\cdot} \mathbf{A}'} \quad \text{avec} \quad 0 < \delta < \tfrac{1}{2}, \tag{6.6}$$

2. et le problème (6.5c) est résolu exactement,

que l'erreur $\left\| \mathbf{u} - \mathbf{u}_M^{(\xi)} \right\|_2$ est bornée par

$$\frac{1}{1+\delta} \left\| \mathbf{y}_R^{(\xi)} \right\|_{\mathbf{A} \overset{\mathrm{D}}{\cdot} \mathbf{A}'} \leq \left\| \mathbf{u} - \mathbf{u}_M^{(\xi)} \right\|_2 \leq \frac{1}{1-\delta} \left\| \mathbf{y}_R^{(\xi)} \right\|_{\mathbf{A} \overset{\mathrm{D}}{\cdot} \mathbf{A}'}. \tag{6.7}$$

Le respect de la première hypothèse nécessite cependant de choisir le rang R adaptivement (ce qui engendre un coût non négligeable).

Remarque 6.1. *On a présenté ici un algorithme de construction directe de l'approximation de rang M. Aussi, une construction gloutonne peut également être utilisée [Billaud-Friess* et al.*, 2013]. On pourra voir sur la Figure 5.2, les résultats obtenus avec une telle construction gloutonne et un rang $R = 1$ pour les cas tests unidimensionnels utilisés dans le chapitre précédent.*

1. L'évaluation du critère d'arrêt nécessite de pouvoir calculer $\left\| \mathbf{y}^{(\xi)} - \mathbf{y}_R^{(\xi)} \right\|_{\mathbf{A} \overset{\mathrm{D}}{\cdot} \mathbf{A}'}$. La variable auxiliaire $\mathbf{y}^{(\xi)}$ n'étant pas connue, la stratégie proposée est de calculer, pour une valeur de R donné, R' modes supplémentaires de façon à ce que $\mathbf{y}_{R+R'}^{(\xi)} \simeq \mathbf{y}^{(\xi)}$.

Algorithm 5 Construction directe (IMR)PGD

Entrées : $\mathbf{A} = \sum_{m=1}^{M_A} \mathbf{A}_m^S \otimes \mathbf{A}_m^T$, $\mathbf{b} = \sum_{m=1}^{M_b} \boldsymbol{b}_m^S \otimes \boldsymbol{b}_m^T$

Sortie : $\mathbf{u} = \sum_{m=1}^{M} \boldsymbol{w}_m \otimes \boldsymbol{\lambda}_m$

Paramètres : $M, R, \xi_{\max}, \epsilon_{\max}, \xi_{\max}^{\text{aux}}, \epsilon_{\max}^{\text{aux}}, \xi_{\max}^{\text{svd}}, \epsilon_{\max}^{\text{svd}}$

1: $\mathbf{u}_M = \mathbf{0}\ (\in \mathrm{R}_M)$
2: **for** $\xi = 1$ to $\xi_{\max}$
3: $\quad \mathbf{r} = \mathbf{b} - \mathbf{A}\overset{\mathrm{D}}{\cdot}\mathbf{u}_M$
4: $\quad \mathbf{y}_R =$ **(G)PGD-P**$(\mathbf{A}\overset{\mathrm{D}}{\cdot}\mathbf{A}', \mathbf{r}, R, \xi_{\max}^{\text{aux}}, \epsilon_{\max}^{\text{aux}})$
5: $\quad \mathbf{u}_M =$ **SVD**$(\mathbf{u}_M + \mathbf{A}'\overset{\mathrm{D}}{\cdot}\mathbf{y}_R, M, \xi_{\max}^{\text{svd}}, \epsilon_{\max}^{\text{svd}})$
6: $\quad$ Verifier la convergence de $\mathbf{u}_M$ par rapport à $\epsilon_{\max}$
7: **end**
8: $\mathbf{u} = \mathbf{u}_M$

Exemple 6.1. (Complexité de l'algorithme IMR-PGD) Dans cet exemple, on décrit la complexité de l'algorithme (IMR)PGD introduit dans ce chapitre.

L'étape la plus coûteuse est l'étape (4) associée à la construction d'une approximation de rang R de la solution courante du problème auxiliaire. On utilise ici une construction gloutonne pure et le coût de cette étape est donc de l'ordre de $\xi_{\max}^{\text{aux}} R(\mathbf{lin}(n_S) + \mathbf{lin}(n_T))$.

Vient ensuite l'étape (5) qui peut être vue comme une projection de l'itéré courant $(\mathbf{u}_M + \mathbf{A}'\overset{\mathrm{D}}{\cdot}\mathbf{y}_R)$ sur le sous-ensemble R_M. Cette projection étant définie au sens de la norme $\|\ \|_2$, elle peut être résolue à l'aide d'un processus de minimisations alternées qui ne nécessite de calculer que des produits scalaires. Le terme $\mathbf{u}_M + \mathbf{A}'\overset{\mathrm{D}}{\cdot}\mathbf{y}_R$ étant donné sous format séparé, cette étape est peu coûteuse en comparaison de l'étape (4). On utilise ici un algorithme de construction gloutonne pure qui nécessite de l'ordre de $\xi_{\max}^{\text{svd}} M(M + M_A R)(n_S + n_T)$ opérations.

L'Algorithme 5 est associé à une construction directe d'une approximation de $\mathbf{u}$ dans R_M (on minimise par rapport à tous les modes dans chaque dimension). Cependant, cette algorithme nécessite seulement de résoudre des systèmes linéaires de taille $n_S \times n_S$ ou $n_T \times n_T$ (et non pas de taille $Mn_S \times Mn_S$ ou $Mn_T \times Mn_T$ comme c'était le cas pour l'algorithme PGD-S introduit dans le chapitre précédent). Aussi, un avantage important de cet algorithme est qu'il peut être utilisé même pour de grandes valeurs du rang M.

6.2.2 Extension pour les problèmes multi-champs

Dans le cas de problèmes multichamps et avec le formalisme introduit au Chapitre 4, l'algorithme s'écrit simplement : connaissant $[\mathbf{u}]_M^{(\xi)} \in \mathrm{R}_{F,M}$, trouver $[\mathbf{u}]_M^{(\xi+1)} \in \mathrm{R}_{F,M}$ tel que

$$[\mathbf{r}]^{(\xi)} = [\mathbf{b}] - [\![\mathbf{A}]\!] \overset{\mathrm{D}}{\cdot} [\mathbf{u}]_M^{(\xi)}, \tag{6.8a}$$

$$[\mathbf{y}]_R^{(\xi)} = \textbf{(G)PGD-P}([\![\mathbf{A}]\!] \overset{\mathrm{D}}{\cdot} [\![\mathbf{A}]\!]', [\mathbf{r}]^{(\xi)}, R), \tag{6.8b}$$

$$[\mathbf{u}]_M^{(\xi+1)} = \arg \min_{[\mathbf{u}]^\star \in \mathrm{R}_{F,M}} \left\| [\mathbf{u}]_M^{(\xi)} + [\![\mathbf{A}]\!]' \overset{\mathrm{D}}{\cdot} [\mathbf{y}]_R^{(\xi)} - [\mathbf{u}]^\star \right\|_2, \tag{6.8c}$$

où l'opérateur **(G)PGD-P**$([\![\mathbf{A}]\!], [\mathbf{b}], M)$ donne une approximation de rang M de $[\![\mathbf{A}]\!]^{-1} \overset{\mathrm{D}}{\cdot} [\mathbf{b}]$, calculée avec l'algorithme glouton de la PGD de type Galerkin.

6.3 Application au problème d'élastodynamique 2D

Dans cette section, on évalue la capacité de l'algorithme (IMR)PGD à calculer une bonne approximation de la meilleure approximation de rang M au sens de la norme $\| \|_2$. L'algorithme est appliqué sur le problème d'élastodynamique bidimensionnel présenté dans l'Exemple 4.3. Les résultats sont principalement extraits de la publication [Boucinha *et al.*, 2013a].

Description du cas test

Le cas test est décrit en détails dans l'Exemple 4.3. On rappelle que le problème espace-temps est formulé avec la méthode de Galerkin discontinue en temps à deux champs (déplacement-vitesse). Le problème étant bidimensionnel, les champs de déplacement et de vitesse sont des champs de vecteurs. Aussi, le problème espace-temps est vu comme un problème à quatres champs, à savoir $u_1(\boldsymbol{x},t), u_2(\boldsymbol{x},t), v_1(\boldsymbol{x},t)$ et $v_2(\boldsymbol{x},t)$, où u_1, u_2 sont les composantes du vecteur déplacement et v_1, v_2 celles du vecteur vitesse. On note $\mathbf{U}_1$, $\mathbf{U}_2$, $\mathbf{V}_1$ et $\mathbf{V}_2$ les représentations discrètes de ces champs [2], et $\mathbf{U}_{1M}$, $\mathbf{U}_{2M}$, $\mathbf{V}_{1M}$ et $\mathbf{V}_{2M}$ leurs approximations de rang M dans R_M. Le problème est discrétisé en espace et en temps avec des éléments finis linéaires (discontinus en temps). Les conditions aux limites de Dirichlet et les conditions initiales en déplacement-vitesse sont prises homogènes. Une condition de Neumann de type choc est appliquée sur l'un des bords de la structure. Afin de caractériser différents régimes dynamiques transitoires, la durée de ce choc est choisie en fonction du nombre sans dimension κ défini de façon similaire au cas d'un problème unidimensionnel. Les résultats sont présentés pour des valeurs de κ égales à $0.03 - 0.7 - 12 - 298$. Enfin pour chaque valeur de κ, les paramètres du maillage espace-temps sont choisis de façon à ce que l'erreur de discrétisation (sur tout le domaine espace-temps et évaluée avec

2. Tous les champs sont approchés dans l'espace d'approximation $\mathscr{U}_h^S \otimes \mathscr{U}_{\Delta t}^{\check{T}}$ avec $\dim(\mathscr{U}_h^S) = n_S$ et $\dim(\mathscr{U}_{\Delta t}^{\check{T}}) = n_T$.

le champ de déplacement) soit inférieur à 4% pour tous les cas tests. Les solutions obtenues pour les différentes valeurs de κ sont représentées dans le domaine espace-temps sur la Figure 6.1.

6.3.1 Décomposition a posteriori

On évalue tout d'abord l'efficacité d'une représentation à variables séparés espace-temps en terme de gain mémoire, dans le cas de ce problème bidimensionnel. Ayant résolu le problème espace-temps dans une première étape, on calcule a posteriori les meilleures approximations de rang M (au sens de la norme $\|\ \|_2$) des composantes $\mathbf{U}_1$, $\mathbf{U}_2$ et $\mathbf{V_1}, \mathbf{V}_2$ des vecteurs déplacement et vitesse. On définit alors l'erreur due à une approximation de rang M (par rapport à la solution discrète) de la façon suivante

$$\text{err}(\mathbf{U}_{1M}, \mathbf{U}_{2M}) = \frac{\sqrt{\|\mathbf{U}_1 - \mathbf{U}_{1M}\|_2^2 + \|\mathbf{U}_2 - \mathbf{U}_{2M}\|_2^2}}{\sqrt{\|\mathbf{U}_1\|_2^2 + \|\mathbf{U}_2\|_2^2}}, \tag{6.9}$$

dans le cas du champ de déplacement, et on note de façon similaire err($\mathbf{V}_{1M}, \mathbf{V}_{2M}$) dans le cas du champ de vitesse. L'erreur obtenue en fonction du rang M est présentée pour tous les cas tests sur la Figure 6.1. En suivant l'approche proposée dans le Chapitre 2, on cherche alors le rang $M_{\min}$ tel que l'erreur due à une approximation de ce rang, soit inférieure à l'erreur de discrétisation. On évalue alors le gain mémoire par

$$\text{gain mémoire} = \frac{n_S \times n_T}{M_{\min} \times (n_S + n_T)}, \tag{6.10}$$

où n_S et n_T sont les dimensions de l'espace d'approximation tels que l'erreur de discrétisation soit égale à 4%. Les gains obtenus pour les différents cas tests sont reportés dans le Tableau 6.1. Même si le rang $M_{\min}$ augmente avec la valeur de κ, on observe que le gain mémoire est d'autant meilleur que la valeur de κ est grande. Ce qui s'explique par l'augmentation plus importante de la dimension de l'espace d'approximation (nécessaire pour obtenir une erreur de discrétisation donnée) lorsque κ augmente.

κ	n_S	n_T	$M_{\min}$	**Gain mémoire**
0.03	40	80	2	**13**
0.7	544	320	6	**34**
12	8320	1280	32	**35**
298	51520	3200	81	**37**

TABLE 6.1: Gain mémoire obtenu avec la meilleure approximation de rang $M_{\min}$ du champ de déplacement.

Aussi, une représentation à variables séparées espace-temps permet de réduire l'espace mémoire nécessaire au stockage de la solution de ce problème bidimentionnel de propagation d'ondes. La question est maintenant de savoir si l'algorithme

(IMR)PGD introduit dans ce chapitre permet de calculer une bonne approximation de la meilleure approximation de rang M (au sens de la norme $\|\ \|_2$) sans avoir à calculer un grand nombre de mode espace-temps.

6.3.2 Décomposition a priori quasi-optimale

On compare ici les approximations de rang M obtenues d'une part avec l'algorithme (R)PGD-S présenté dans le chapitre précédent et d'autre part avec l'algorithme (IMR)PGD présenté dans ce chapitre, à la meilleure approximation de rang M (calculée a posteriori et définie par rapport à la norme $\|\ \|_2$).

Influence des paramètres de discrétisation

On compare tout d'abord la précision de l'approximation, à un rang $M = 10$ fixé, pour le cas test avec $\kappa = 0.7$. Les résultats sont regroupés dans le Tableau 6.2. Comme dans le cas unidimensionnel présenté dans le chapitre précédent, on observe que l'algorithme (R)PGD-S donne une approximation non-optimale et peu précise par rapport à la meilleure approximation donnée par la SVD. De plus, la précision de l'approximation (R)PGD-S se détériore rapidement lorsque la dimension de l'espace d'approximation augmente. On observe au contraire, que l'algorithme (IMR)PGD calcule une très bonne approximation de la meilleure approximation donnée par la SVD, et ce pour le champ de déplacement et de vitesse. Même pour une valeur du rang auxiliaire R relativement faible (voir l'erreur pour $R = 5$), l'approximation obtenue est quasi-optimale. De plus, la précision de l'approximation (IMR)PGD est beaucoup moins affectée par l'augmentation de la dimension de l'espace d'approximation.

Convergence de l'algorithme (IMR)PGD

Le paramètre clé de l'algorithme (IMR)PGD, tant en terme de précision que de coût de calcul, est le rang R de l'approximation de la solution du problème auxiliaire. Aussi, on compare ici la précision de l'approximation obtenue pour différentes valeurs du rang R (on prend $R = 1-2-4-8-16$) et pour les cas tests à $\kappa = 0.03-0.7-12$ (pour lesquels l'erreur de discrétisation est fixée à 4%). Pour chaque valeur de R, on calcule la décomposition de rang $M = 1-2-4-8-16-32-64$ avec l'algorithme (IMR)PGD (chaque décomposition de rang M est calculée indépendemment l'une de l'autre). Les résultats obtenus sont présentés sur la Figure 6.2. Pour tous les cas tests et toutes les valeurs de M choisies, on observe que l'algorithme (IMR)PGD permet de trouver une très bonne approximation de la décomposition optimale avec un très faible nombre de modes auxiliaires. Par exemple, on obtient une très bonne approximation de la meilleure approximation de rang $M = 64$ pour le cas test $\kappa = 12$ avec seulement un mode auxiliaire.

Cependant, la convergence de l'algorithme dépend beaucoup du choix de R et il faut d'autant plus d'itérations pour converger que la valeur de R est faible. Un com-

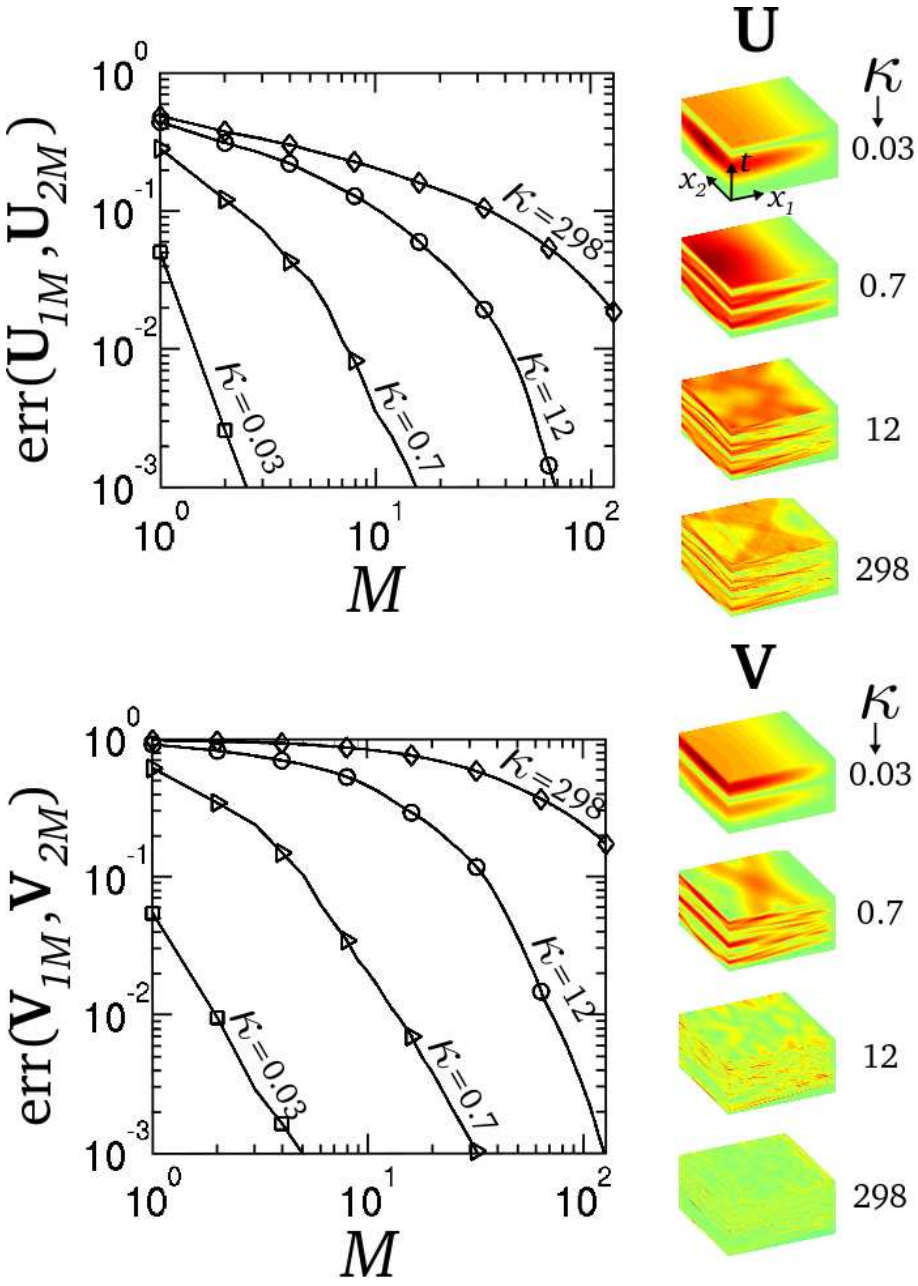

FIGURE 6.1: Erreur due à une approximation de rang M des composantes des champs de déplacement et de vitesse pour les cas tests de l'Exemple 4.3.

$n_S \times n_T$	err($\mathbf{U}_{1M}, \mathbf{U}_{2M}$) pour $M = 10$					
	SVD	(R)PGD-S	(IMR)PGD			
			R=5	R=10	R=20	R=40
144 × 160	2.5e-03	3.5e-02	3.4e-03	2.9e-03	2.6e-03	2.5e-03
544 × 320	3.5e-03	3.9e-02	4.8e-03	3.9e-03	3.6e-03	3.5e-03
2112 × 640	3.6e-03	9.8e-02	4.4e-03	4.0e-03	3.8e-03	3.7e-03
8320 × 1280	3.6e-03	1.0e+00	4.4e-03	4.0e-03	3.7e-03	3.6e-03
$n_S \times n_T$	err($\mathbf{V}_{1M}, \mathbf{V}_{2M}$) pour $M = 10$					
	SVD	(R)PGD-S	(IMR)PGD			
			R=5	R=10	R=20	R=40
144 × 160	1.4e-02	1.3e-01	1.8e-02	1.6e-02	1.5e-02	1.4e-02
544 × 320	2.0e-02	1.4e-01	2.7e-02	2.4e-02	2.1e-02	2.1e-02
2112 × 640	2.2e-02	3.1e-01	3.0e-02	2.6e-02	2.3e-02	2.2e-02
8320 × 1280	2.1e-02	9.7e-01	4.5e-02	3.2e-02	2.4e-02	2.2e-02

TABLE 6.2: Erreur due à une approximation de rang $M = 10$ en fonctions de la dimension de l'espace d'approximation, pour différentes décompositions et le cas test $\kappa = 0.7$.

promis doit donc être trouvé entre le choix du paramètre R et le nombre d'itérations nécessaires à l'algorithme (IMR)PGD pour converger. Une étude détaillée a été réalisée à ce sujet dans [Boucinha *et al.*, 2013a] et n'est pas reportée ici. On retiendra que la vitesse de convergence de l'algorithme dépend du nombre de modes auxiliaires R, et que pour une valeur de R fixée, il faut d'autant plus d'itérations (pour atteindre la convergence) que le rang M de l'approximation que l'on cherche est grand. Typiquement, pour le cas test $\kappa = 12$, la décomposition de rang $M = 16$ est obtenue en $\xi_{\max} \simeq 30$ itérations en choisissant $R = 1$ alors que seulement $\xi_{\max} \simeq 5$ itérations sont nécessaires avec le choix $R = 16$. Encore avec ce choix ($R = 16$), il faut $\xi_{\max} \simeq 10$ itérations pour obtenir la décomposition de rang $M = 32$.

Un autre point concerne le critère d'arrêt utilisé pour stopper les itérations. Comme on ne peut pas évaluer exactement la fonctionnelle que l'on minimise (qui n'est autre que $\left\| [\mathbf{u}] - [\mathbf{u}]_M \right\|_2$), le choix du critère d'arrêt est moins naturelle que pour les algorithmes PGD présentés dans le chapitre précédent. On peut cependant utiliser la valeur de $J_R^\xi = \left\| [\mathbf{y}]_R^{(\xi)} \right\|_{[\![\mathbf{A}]\!]^{\mathrm{D}}, [\![\mathbf{A}]\!]'}$ qui donne une bonne estimation de $\left\| [\mathbf{u}] - [\mathbf{u}]_M^{(\xi)} \right\|_2$ si le rang R est suffisamment grand (voir la Figure 6.3). Néanmoins, lorsque le rang R est faible, la valeur de J_R^ξ est très oscillante et il est difficile de détecter la stagnation de cette fonctionnelle. Ces oscillations peuvent être attenuées en choisissant le rang R de manière adaptative comme proposée par [Billaud-Friess *et al.*, 2013], mais il n'existe pas (à ce jour) de procédure efficace pour choisir le rang R de manière adaptative.

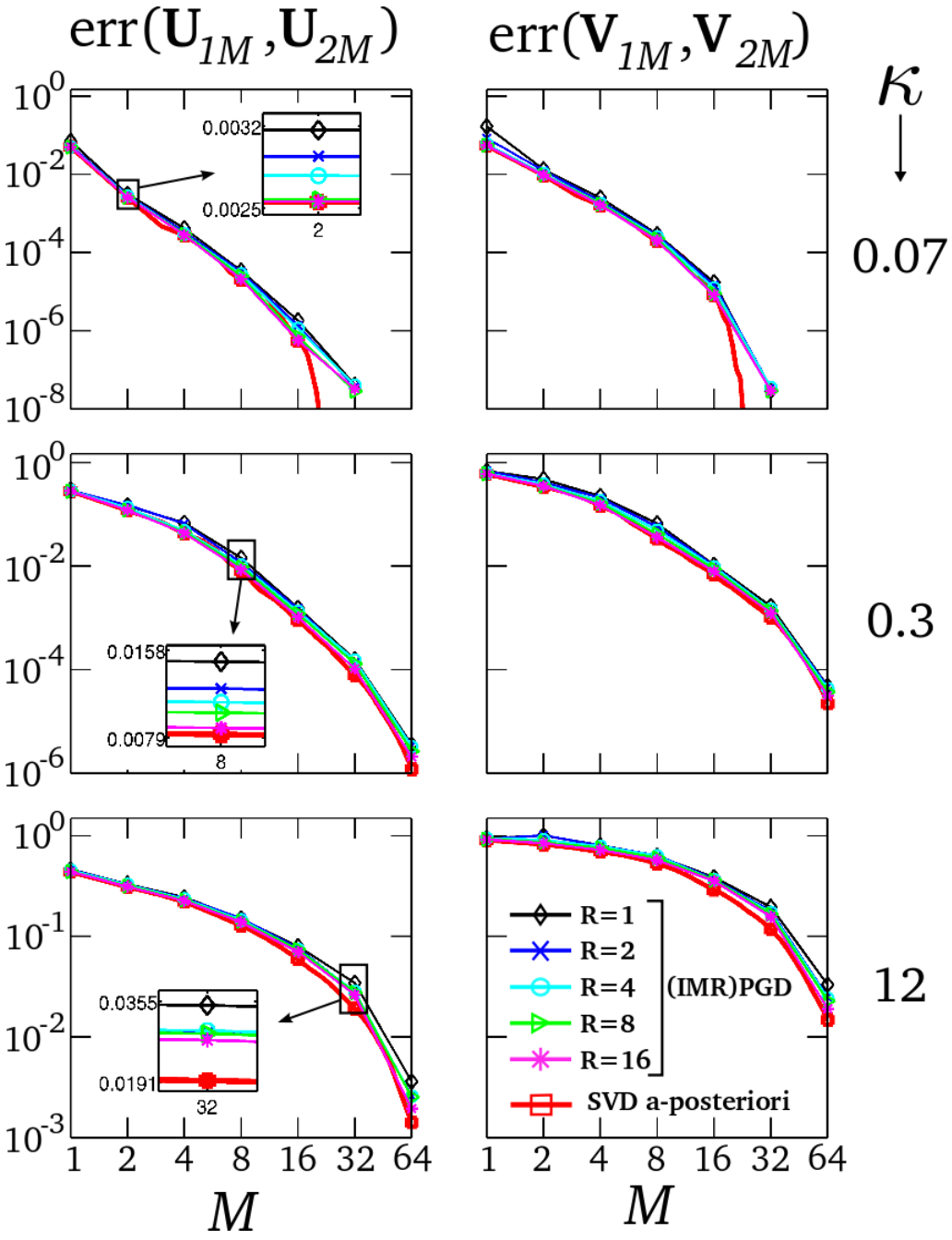

FIGURE 6.2: Erreur due à une approximation de rang M calculée avec l'algorithme (IMR)PGD en fonction du rang auxiliaire R et comparaison avec la SVD.

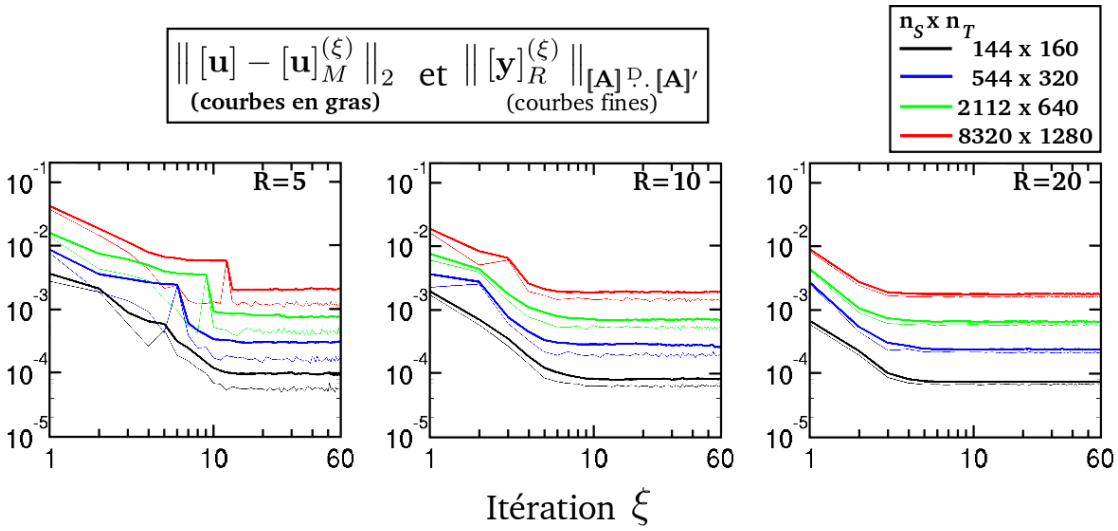

FIGURE 6.3: Estimation de l'erreur due à une approximation de rang M (donnée par $\| [\mathbf{u}] - [\mathbf{u}]_M^{(\xi)} \|_2$) avec la fonctionnelle $J_R^\xi = \| [\mathbf{y}]_R^{(\xi)} \|_{[\![\mathbf{A}]\!] \ddot{\cdot} [\![\mathbf{A}]\!]'}$ en fonction du rang R et des paramètres de discrétisation (pour le cas test avec $\kappa = 0.7$).

6.4 Conclusion

Dans ce chapitre, une nouvelle approche, récemment introduite par [Billaud-Friess *et al.*, 2013] a été présentée. Cette approche permet de calculer *a priori* une approximation de faible rang quasi-optimale (au sens d'une norme choisie) de la solution d'un problème donné sous format tensoriel. Cette approche est basée sur la perturbation d'un problème de minimisation dans R_M d'une norme idéale du résidu. Un algorithme, appelé (IMR)PGD (« Ideal Minimal Residual PGD »), a été implémenté dans le cadre d'un problème générique à $F-$champs. L'efficacité de cet algorithme a été illustrée dans le cas d'un problème académique de propagation d'ondes dans un milieu bidimensionnel. Pour le problème traité, cet algorithme a permis d'obtenir une très bonne approximation de la meilleure approximation de rang M au sens de la norme $\| \|_2$.

La précision de l'approximation obtenue dépend d'une approximation de faible rang de la solution d'un problème auxiliaire. Les résultats numériques ont montrés que l'algorithme développé converge vers une approximation quasi-optimale même pour de très faibles valeurs du rang auxiliaire, témoignant du fait que la solution du problème auxiliaire peut être approchée précisémment avec très peu de modes espace-temps. Cependant, la vitesse de convergence de l'algorithme dépend fortement du nombre de modes auxiliaires utilisés, et un plus grand nombre de modes auxiliaires doivent être utilisés pour diminuer le nombre d'itérations nécessaires à l'algorithme pour converger. On notera également que l'approximation de la solution du problème auxiliaire peut être utilisée pour calculer une bonne estimation de l'erreur due à une approximation de rang M, si le rang auxiliaire est suffisament grand.

Conclusions et perspectives

Dans ce manuscrit, une stratégie de calcul innovante a été appliquée à la résolution de problèmes de dynamique transitoire. Cette stratégie repose sur la construction d'une approximation à variables séparées espace-temps, à l'aide d'un solveur non-incrémental exploitant la structure tensorielle des opérateurs du problème espace-temps. En notant n_S et n_T les dimensions des espaces d'approximation spatiale et temporelle respectivement, la stratégie permet de remplacer les $n_S n_T$ valeurs nécessaires au stockage d'une approximation classique d'un champ défini sur le domaine espace-temps, par le stockage de $M(n_S + n_T)$ valeurs dans le cas d'une approximation à variables séparées espace-temps, où M est un nombre de modes espace-temps. Une résolution incrémentale d'un problème transitoire nécessite de résoudre n_T fois un système linéaire de taille $n_S \times n_S$. Le solveur développé permet d'obtenir une approximation à variables séparées espace-temps en $M\xi$ résolutions alternatives d'un système linéaire de taille $n_S \times n_S$ et d'un autre de taille $n_T \times n_T$, où ξ est un nombre d'itérations.

L'efficacité de cette stratégie a été évaluée pour des problèmes académiques de dynamique transitoire (de type propagation d'ondes dans un milieu uni- et bidimensionnel consécutive à un choc). Les résultats obtenus montrent qu'une représentation à variables séparées espace-temps permet d'approcher présicemment la solution des problèmes traités (dans le domaine basse et moyenne fréquence). Le nombre de modes espace-temps, nécessaires pour obtenir une approximation aussi précise que la solution discrète du problème, est suffisamment faible pour que la représentation à variables séparées espace-temps permette de réduire la mémoire nécessaire au stockage de la solution sur le domaine espace-temps, de plus d'un ordre de grandeur. La géométrie des cas étudiés étant très simplifiée, des gains encore plus significatifs devraient être obtenus dans le cas de géométries complexes. L'approximation *a posteriori* d'un champ sous la forme d'une représentation à variables séparées (espace-temps ou plus générale) apparaît donc comme un outil de compression de données très efficace, ouvrant la voie au développement d'outils de post-traitement intelligents, qui incorporent, en plus de leur fonctionnalité de visualisation des résultats, des outils de calculs et de représentation de données dans ce format innovant.

Dans le cas d'une unique résolution d'un problème transitoire, l'efficacité du solveur développé est plus discutable. Un état de l'art des algorithmes existants a permis d'évaluer l'efficacité des définitions classiques de la PGD dans le cas des prob-

lèmes académiques de dynamique transitoire mentionnés. Les résultats obtenus ont mis en défaut l'optimalité de la PGD la plus robuste. Le nombre de modes espace-temps nécessaires pour obtenir une approximation suffisamment précise de la solution est alors trop grand pour que le solveur permette de réduire le temps de calcul. Aussi, nos efforts se sont portés sur le développement d'un nouvel algorithme, proposé par [Billaud-Friess *et al.*, 2013], et permettant d'obtenir une approximation à variables séparées quasi-optimale sans avoir à calculer un grand nombre de modes espace-temps . Les résultats obtenus avec ce dernier solveur ont montré qu'il est possible de calculer a priori une très bonne approximation de la meilleure approximation à variables séparées au sens d'une norme choisie à l'avance. Ces derniers résultats permettent d'envisager de nombreuses perspectives.

Disposer d'un solveur non-incrémental efficace, permet tout d'abord d'envisager le développement de nouveaux espaces d'approximation temporelle, dont la conception ne soit pas limitée par le caractère incrémental des stratégies classiquement utilisées pour résoudre des problèmes transitoires. Des méthodes spectrales en espace et en temps sont par exemple tout à fait envisageables si l'on dispose d'un solveur efficace pour résoudre le problème espace-temps [Dumon *et al.*, 2013].

L'efficacité d'un solveur basé sur la PGD doit être comparée à la méthode de référence utilisée pour résoudre le problème considéré. Dans le cas d'un problème de dynamique transitoire, discrétisé avec les méthodes classiques d'approximation, la méthode de référence est la stratégie incrémentale. Ces méthodes d'approximation ayant été developpées pour permettre une telle résolution, le problème espace-temps obtenu, a une structure particulière (tridiagonale inférieure) qui rend sa résolution peu naturelle avec un solveur itératif. Aussi, le solveur PGD doit être utilisé dans le cas où il n'existe pas de stratégie de résolution efficace pour résoudre le problème de référence. Pour réaliser une étude paramétrique, on souhaitera par exemple simuler n_p fois un problème transitoire (où n_p est la dimension de l'espace paramétrique) et la méthode de référence nécessitera une complexité de l'ordre de $n_p n_T \mathbf{lin}(n_S)$ opérations (où $\mathbf{lin}(n)$ est la complexité associée à la résolution d'un système linéaire de taille $n \times n$). Dans ce cas, un solveur PGD permettra d'obtenir une approximation à variables séparées espace-temps-paramètre avec une complexité de l'ordre de $M\xi(\mathbf{lin}(n_S) + \mathbf{lin}(n_T) + \mathbf{lin}(n_p))$ et on peut raisonnablement espérer qu'il permette de réduire le temps de calcul dans ce cas [Chinesta *et al.*, 2013]. Un autre exemple concerne le développement d'éléments finis structuraux à comportement volumique pour des applications de dynamique transitoire. Pour de nombreuses géométries d'intérêt, on peut construire une approximation à variables séparées espace-espace [Bognet *et al.*, 2012]. Dans le cas d'une application en dynamique transitoire, la complexité de la méthode de référence est alors de l'ordre de $n_T \mathbf{lin}(n_{S1} n_{S2})$ (où n_{S1} et n_{S2} sont les dimensions des espaces d'approximation définies sur des domaines spatials Ω_1 et Ω_2 respectivement). Dans ce cas, le solveur PGD donnera une approximation en $M\xi(\mathbf{lin}(n_{S1}) + \mathbf{lin}(n_{S2}) + \mathbf{lin}(n_T))$ et on peut de nouveau espérer une diminution du temps de calcul.

Ces exemples nécessitent cependant de calculer une approximation de faible rang d'un tenseur d'ordre élevé ($D \geq 3$). Dans ce cas, le problème de minimisation dans le sous-ensemble des décompositions canoniques de rang $M > 1$ est mal posé [De Silva et Lim, 2008]. En pratique, ceci se traduit par des difficultés numériques pour construire une approximation d'un tenseur dans ce sous ensemble. La stratégie de construction gloutonne pure permet d'éviter ces problèmes. Cependant, elle nécessite des mises à jour très coûteuse pour améliorer la précision de l'approximation de rang M. Aussi, d'autres sous-ensembles de tenseurs (tels que les sous-ensembles de Tucker ou de Tucker hiérarchique [Hackbusch, 2012]) doivent être considérés dans le cadre de stratégies de construction a priori d'une approximation de faible rang d'un tenseur d'ordre élevé. Dans cet esprit, la stratégie de projection sur un espace réduit obtenu par construction gloutonne, proposée par [Giraldi, 2012], s'est révélée très efficace et des développements dans cette direction mériteraient d'être entrepris.

Enfin, les résultats obtenus dans ce manuscrit laissent à penser qu'une stratégie de résolution multigrille espace-temps permettrait de réduire considérablement la complexité associée à la construction d'une approximation à variables séparés espace-temps. Une telle stratégie a notamment été appliquée avec succès par [De Sterck et Miller, 2013] dans le cas d'une construction a posteriori. Classiquement, un solveur multigrille nécessite trois ingrédients principaux qui sont [Hackbusch, 2003] :

- une hiérarchie de systèmes linéaires,
- des opérateurs de changement d'échelle (sur les quantités primales et duales),
- et un algorithme itératif, dit « lisseur », qui doit posséder la propriété de réduire les erreurs hautes fréquences en quelques itérations.

Dans le cadre d'une approximation à variables séparées espace-temps, la construction d'une hiérarchie de grilles est facilitée par la nature tensorielle de l'approximation. La construction d'opérateurs de changement d'échelle ne pose pas non plus de problème particulier (on peut définir un opérateur de changement d'échelle espace-temps par tensorisation d'opérateurs de changement d'échelle en espace et en temps). Le choix d'un lisseur est plus délicat. En effet, le sous-ensemble des décompositions canoniques de rang $M > 1$ n'est pas un espace vectoriel. Aussi, l'utilisation d'un lisseur classique augmente le rang du nouvel itéré, et celui-ci doit être tronqué d'une manière ou d'une autre. Des simulations ont été menées dans le cadre d'une telle stratégie, en choisissant le solveur PGD comme lisseur. Les résultats préliminaires ont montré que l'erreur introduite par la troncature de la correction de l'itéré courant détériore la vitesse de convergence de l'algorithme PGD.

Néanmoins, l'initialisation de l'algorithme PGD à l'aide de la prolongation d'une décomposition calculée sur une grille grossière a permis de diminuer le nombre d'itérations ξ nécessaires pour que le processus de minimisations alternées converge. Le développement de stratégies basées sur ce principe permettrait donc d'accélérer

la convergence du processus de minimisations alternées. Un autre aspect concerne le choix des espaces d'approximation pour les différents modes en espace et en temps. Il s'avère en effet inutile de choisir le même espace d'approximation pour tous les modes. Une représentation à variables séparées où chaque mode serait cherché dans un espace d'approximation différent permettrait ainsi de réduire considérablement les coûts de calcul.

Annexe A

Notations & Opérations algébriques

On décrit, dans cette annexe, les principales notations utilisées dans le manuscrit, ainsi que les règles de calcul entre les différents objets mathématiques manipulés.

Sommaire

A.1 Notations

Les principales notations sont résumées dans le Tableau A.1. On respecte globalement les conventions suivantes :

- les scalaires sont notés en *italique* ou classiquement : $a, b, A, B, \phi, \psi, \cdots$,a,b,$\cdots$,
- les vecteurs (tenseurs d'ordre 1) sont notés en ***italique gras*** : $\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{A}, \boldsymbol{B}, \boldsymbol{\phi}, \boldsymbol{\psi}, \cdots$,
- les tenseurs d'ordre $D \geq 2$ sont notés en **gras** : $\mathbf{a}, \mathbf{b}, \mathbf{A}, \mathbf{B}, \mathbf{\Phi}, \mathbf{\Psi}, \cdots$.

Symbole	Désignation	Symbole	Désignation
a	Scalaire	$[\![\,]\!]$	$F \times F-$ tuple
$\boldsymbol{a}$	Vecteur	$u(x,t)$	Champ scalaire
$\mathbf{a}$	Tenseur d'ordre $D \geq 2$	$\boldsymbol{u}(\boldsymbol{x},t)$	Champ vectoriel
$[\,]$	$F-$ tuple	$\mathscr{U}(\Omega)$	Espace fonctionnel

TABLE A.1: Principales notations

A.1.1 Vecteurs & Tenseurs d'ordre D

On se reporte à l'ouvrage récent de [Hackbusch, 2012] pour une définition détaillée du concept d'espace produit tensoriel. On se contente ici de la définition suivante : étant donnés D espaces vectoriels, $E^1, \cdots, E^D$ de dimensions finies avec $\dim(E^d) = n_d$ pour $d = 1, \cdots, D$, on peut définir le *produit tensoriel* $\otimes$ et l'*espace produit tensoriel* $E^1 \otimes \cdots \otimes E^D$ comme suit :

Définition A.1. (Produit tensoriel & Espace produit tensoriel & Tenseur) Soit une application $\otimes : E^1 \times \cdots \times E^D \to E$ vérifiant les propriétés suivantes :

1. $\otimes$ est multilinéaire,
2. si les vecteurs $\boldsymbol{e}_1^d, \cdots, \boldsymbol{e}_{n_d}^d$ forment une famille libre de E^d pour $d = 1, \cdots, D$, alors les éléments $\otimes(\boldsymbol{e}_{i_1}^1, \cdots, \boldsymbol{e}_{i_D}^D)$ avec $i_d = 1, \cdots, n_d$ pour $d = 1, \cdots, D$ forment une famille libre.

Alors l'application $\otimes$ est appelé *produit tensoriel*. Un élément $\boldsymbol{e}^1 \otimes \cdots \otimes \boldsymbol{e}^D = \otimes(\boldsymbol{e}^1, \cdots, \boldsymbol{e}^D)$ est appelé un *tenseur élémentaire*. L'espace $E = E^1 \otimes \cdots \otimes E^D$, défini par

$$E = \operatorname{span}\left\{ \otimes(\boldsymbol{e}^1, \cdots, \boldsymbol{e}^D) ; \boldsymbol{e}^d \in E^d \text{ pour } d = 1, \cdots, D \right\}, \tag{A.1}$$

est appelé *espace produit tensoriel* et $\dim(E) = \prod_{d=1}^{D} \dim(E^d) = \prod_{d=1}^{D} n_d$. Un élément $\mathbf{v} \in E$ est appelé un *tenseur* et peut s'écrire sous la forme :

$$\mathbf{v} = \sum_{i_1=1}^{n_1} \cdots \sum_{i_D=1}^{n_D} \mathrm{v}_{i_1 \cdots i_D} \boldsymbol{e}_{i_1}^1 \otimes \cdots \otimes \boldsymbol{e}_{i_D}^D . \tag{A.2}$$

Le choix pour les espaces E^d n'est pas fixé a priori. **Dans ce manuscrit, on adopte le point de vue de l'implémentation** : on considère les vecteurs comme des tableaux à une dimension, et de façon générale, les tenseurs d'ordre D comme des tableaux à D dimensions [1]. Plus précisément, on désigne par vecteur, un élément de $\mathbb{R}^n$ et par tenseur d'ordre D, un élément de $\bigotimes_{d=1}^{D} \mathbb{R}^{n_d}$. Bien sûr, on peut également considérer un élément d'un espace de fonctions (par exemple le champ scalaire de déplacement $u(x,t)$ défini sur le domaine espace-temps $\Omega \times I$) comme un vecteur ou un tenseur. Cependant, pour simplifier les notations, on réserve les appellations vecteur ou tenseur aux coordonnées de ce champ dans une base de l'espace fonctionnel auquel il appartient. Par exemple, le champ scalaire $u(x,t)$ appartenant à l'espace produit tensoriel $\mathscr{U}(\Omega \times I) = \mathscr{U}^S(\Omega) \otimes \mathscr{U}^T(I)$ peut être vu comme un tenseur d'ordre deux. En effet, en supposant que les espaces $\mathscr{U}^S$ et $\mathscr{U}^T$ sont de dimensions finies (notées n_S et n_T respectivement), on peut exprimer ce champ sur une base de $\mathscr{U}$ par tensorisation des bases $\left(\phi_1, \cdots, \phi_{n_S}\right)$ de $\mathscr{U}^S$ et $\left(\psi_1, \cdots, \psi_{n_T}\right)$ de $\mathscr{U}^T$, comme suit :

$$u(x,t) = \sum_{i=1}^{n_S} \sum_{j=1}^{n_T} u_{ij}\, \phi_i(x) \otimes \psi_j(t)\,. \tag{A.3}$$

Dans ce cas, le produit tensoriel est défini sur $\mathscr{U}^S(\Omega) \times \mathscr{U}^T(I)$. Il est équivalent à la multiplication classique des fonctions ϕ_i et ψ_j, c'est-à-dire $\phi_i(x) \otimes \psi_j(t) = \phi_i(x)\psi_j(t)$. Le champ $u(x,t)$ est, de ce point de vue, un tenseur d'ordre deux et devrait être, selon la convention du Tableau A.1, écrit en gras. Cependant, dans ce manuscrit, on choisit d'écrire l'équation (A.3) de la façon suivante :

$$u(x,t) = \sum_{i=1}^{n_S} \sum_{j=1}^{n_T} \mathrm{u}_{ij}\, \phi_i(x)\psi_j(t) = \mathbf{u} \overset{\mathrm{D}}{\cdot} (\phi \otimes \psi)\,, \tag{A.4}$$

où le produit scalaire « $\overset{\mathrm{D}}{\cdot}$ » est défini dans la section suivante. Dans ce cas, le produit tensoriel est défini sur $\mathbb{R}^{n_S} \times \mathbb{R}^{n_T}$ et ce sont les coordonnées du champ dans la base de $\mathscr{U}$ (noté $\mathbf{u}$) qui constituent un tenseur d'ordre deux. Le tenseur $\mathbf{u} \in \mathbb{R}^{n_S} \otimes \mathbb{R}^{n_T}$ sera également appelé la représentation discrète du champ $u \in \mathscr{U}^S(\Omega) \otimes \mathscr{U}^T(I)$. C'est cette représentation discrète que l'on cherchera à calculer numériquement.

A.1.2 F-Tuples

Dans le cadre de ces travaux, on sera également amené à utiliser la notion de *tuple*. Un tuple, noté sous forme condensée $[\,]$, est un élément d'un produit d'espaces $E_1 \times \cdots \times E_F$, où chaque espace E_i peut contenir des objets de nature différente de ceux des autres espaces E_j. Un tuple possédant F éléments est appelé F-tuple. On le note sous

1. Un des objectifs de cette thèse est de réduire l'espace mémoire nécessaire au stockage des tableaux à plusieurs dimensions. On cherchera, pour cela, à approximer un tenseur donné, par un tenseur exprimé dans un autre format (sous la forme d'une représentation séparée de faible rang), moins coûteux en espace mémoire.

forme développée, $[a] = [a_1, \cdots, a_F]$ où $a_i \in E_i$ pour $i = 1, \cdots, F$. Deux tuples $[a]$ et $[b]$ d'un même produit d'espaces vérifient les propriétés suivantes :

$$[a] + \alpha[b] = [a_1 + \alpha b_1, \cdots, a_F + \alpha b_F], \tag{A.5}$$

où le tuple $[c] = [a] + \alpha[b]$ appartient également au produit d'espaces considéré. Un exemple simple de tuple est le choix $E_i = \mathbb{R}$ pour $i = 1, \cdots, F$ qui permet de construire des vecteurs de $\mathbb{R}^F$. On donne un exemple plus complexe dans la suite de ce paragraphe. De nombreux langages de programmation permettent de travailler avec des objets de type *tuple*. Dans le langage*Matlab*, un tuple peut être défini avec le type *cell*[2].

Dans ce manuscrit, on utilise la notion de tuple pour simplifier l'écriture des équations associées à la discrétisation des problèmes multi-champs, ou des problèmes faisant intervenir un champ vectoriel. Dans ce dernier cas, **chaque composante d'un champ vectoriel est considérée comme un champ (scalaire) à part entière**. Prenons par exemple, le champ vectoriel de déplacement $\boldsymbol{u}(\boldsymbol{x}, t)$ défini de $\Omega \times I$ dans $\mathbb{R}^3$ et exprimé sur la base canonique $(\boldsymbol{e}_1, \boldsymbol{e}_2, \boldsymbol{e}_3)$ de $\mathbb{R}^3$ comme suit :

$$\boldsymbol{u}(\boldsymbol{x}, t) = u_1(\boldsymbol{x}, t)\boldsymbol{e}_1 + u_2(\boldsymbol{x}, t)\boldsymbol{e}_2 + u_3(\boldsymbol{x}, t)\boldsymbol{e}_3 . \tag{A.6}$$

Alors, chaque composante $u_i(\boldsymbol{x}, t)$ est considérée comme un champ scalaire appartenant à un espace fonctionnel $\mathscr{U}_i(\Omega \times I)$, pour $i = 1, \cdots, 3$. En introduisant une base de $\mathscr{U}_i(\Omega \times I)$, notée (sous format tensoriel) $\phi_i(\boldsymbol{x}) \otimes \psi_i(t)$, la discrétisation du champ $u_i(\boldsymbol{x}, t)$ est donnée par :

$$u_i(\boldsymbol{x}, t) = \mathbf{u}_i \overset{\mathrm{D}}{\cdot} (\phi_i(\boldsymbol{x}) \otimes \psi_i(t)) . \tag{A.7}$$

Alors, on introduira le tuple $[\mathbf{u}] = [\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3]$ pour simplifier l'écriture des équations d'un problème faisant intervenir le champ $\boldsymbol{u}(\boldsymbol{x}, t)$ sous forme discrétisée. L'objet $[\mathbf{u}]$ sera considéré comme la représentation discrète du champ vectoriel $\boldsymbol{u}(\boldsymbol{x}, t)$. C'est un F-tuple (avec $F = 3$) de tenseurs d'ordre D, où chaque tenseur $\mathbf{u}_i$ appartient à l'espace produit tensoriel $\bigotimes_{d=1}^{D} \mathbb{R}^{n(i,d)}$.

Remarque A.1. *Il est à noter que chaque tenseur* $\mathbf{u}_i$ *peut appartenir à un espace de dimension différente : on a* $\dim(\mathscr{U}_i) = \prod_{d=1}^{D} n(i,d)$. *Ceci permettra notamment de formuler un problème d'évolution multi-champs de façon monolithique tout en autorisant une discrétisation différente de chaque champs en espace et en temps, levant ainsi une des principales limitations des formulations monolithiques de ce type de problème [Hübner* et al.*, 2004].*

Remarque A.2. *Avec les conventions de notation adoptées, les composantes* $\mathbf{u}_i$ *d'un* F*-tuple* $[\mathbf{u}]$ *s'obtiennent en supprimant la notation* $[\,]$ *et en ajoutant un indice à* $\mathbf{u}$. *On rappelle que les composantes* $\mathrm{u}_{i_1 \cdots i_D}$ *d'un tenseur* $\mathbf{u}$ *d'ordre D sont obtenues en supprimant la notation en gras et en ajoutant D indices (les composantes* $\mathrm{u}_{i_1 \cdots i_D}$ *sont des scalaires).*

2. Le type *cell* du langage *Matlab* peut être vue comme un tuple dans le sens où il permet de construire un tableau dont chaque case peut contenir un objet de nature différente. Cependant, les opérations algébriques (comme par exemple l'opération de l'équation (A.5)) entre deux objets de type *cell* ne sont pas définies et doivent être programmées explicitement.

A.2 Opérations algébriques

Les principales opérations algébriques sont résumées dans le Tableau A.2. Elles sont détaillées dans la suite de cette section.

Opération	Désignation
$\otimes$	Produit tensoriel
.	Produit scalaire canonique entre vecteurs
:	Produit scalaire canonique entre tenseurs d'ordre 2
$\overset{\mathrm{D}}{\cdot}$	Produit scalaire canonique entre tenseurs d'ordre D
$\overset{\mathrm{D}}{:}$.	Produit scalaire canonique entre F-tuples de tenseurs d'ordre D
$'$	Transposé
$^{-1}$	Inverse

TABLE A.2: Opérations algébriques

A.2.1 Produit scalaire canonique

- Le produit scalaire sur l'espace des vecteurs de $\mathbb{R}^n$ est noté « . » et défini par :

$$\boxed{\boldsymbol{a}.\boldsymbol{b} = \sum_{i=1}^{n} a_i b_i\,, \quad \forall \boldsymbol{a},\boldsymbol{b} \in \mathbb{R}^n}\,. \tag{A.8}$$

- Le produit scalaire sur l'espace $\bigotimes_{d=1}^{D} \mathbb{R}^{n_d}$ des tenseurs d'ordre D est noté « $\overset{\mathrm{D}}{\cdot}$ » (ou également « : » dans le cas $D = 2$) et défini par :

$$\boxed{\mathbf{a} \overset{\mathrm{D}}{\cdot} \mathbf{b} = \sum_{i_1=1}^{n_1} \cdots \sum_{i_D=1}^{n_D} \mathrm{a}_{i_1 \cdots i_D} \mathrm{b}_{i_1 \cdots i_D}\,, \quad \forall \mathbf{a},\mathbf{b} \in \bigotimes_{d=1}^{D} \mathbb{R}^{n_d}}\,. \tag{A.9}$$

- Le produit scalaire sur le produit d'espaces produits tensoriels $E_1 \times \cdots \times E_F$ où chaque $E_i = \bigotimes_{d=1}^{D} \mathbb{R}^{n(i,d)}$ est un espace de tenseurs d'ordre D, est noté « $\overset{\mathrm{D}}{:}$. » et défini par :

$$\boxed{[\mathbf{a}] \overset{\mathrm{D}}{:}. [\mathbf{b}] = \sum_{i=1}^{F} \mathbf{a}_i \overset{\mathrm{D}}{\cdot} \mathbf{b}_i\,, \quad \forall [\mathbf{a}], [\mathbf{b}] \in E_1 \times \cdots \times E_F}\,. \tag{A.10}$$

A.2.2 Système linéaire

- Étant donnés la matrice $\mathbf{A} \in \mathbb{R}^m \otimes \mathbb{R}^n$ et les vecteurs $\boldsymbol{u} \in \mathbb{R}^n$ et $\boldsymbol{b} \in \mathbb{R}^m$, on définit le système d'équations linéaires suivant :

$$\boxed{\mathbf{A}.\boldsymbol{u} = \boldsymbol{b} \quad \Leftrightarrow \quad \sum_{j=1}^{n} \mathrm{A}_{ij}\, u_j = b_i\,, \quad \forall i = 1, \cdots, m}\,. \tag{A.11}$$

- Étant donnés le tenseur $\mathbf{A} \in \bigotimes_{d=1}^{D} \left(\mathbb{R}^{m_d} \otimes \mathbb{R}^{n_d}\right)$, et les tenseurs $\mathbf{u} \in \bigotimes_{d=1}^{D} \mathbb{R}^{n_d}$ et $\mathbf{b} \in \bigotimes_{d=1}^{D} \mathbb{R}^{m_d}$, on définit le système d'équations linéaires suivant :

$$\boxed{\mathbf{A} \overset{\mathrm{D}}{\cdot} \mathbf{u} = \mathbf{b} \quad \Leftrightarrow \quad \sum_{j_1=1}^{n_1} \cdots \sum_{j_D=1}^{n_D} \mathrm{A}_{i_1 j_1 \cdots i_D j_D}\, \mathrm{u}_{j_1 \cdots j_D} = \mathrm{b}_{i_1 \cdots i_D}\,, \quad \begin{cases} \forall i_d = 1, \cdots, m_d \\ \forall d = 1, \cdots, D \end{cases}}\,. \tag{A.12}$$

- Étant donnés le $F \times F$-tuple[3] $[\![\mathbf{A}]\!]$ de tenseurs $\mathbf{A}_{ij} \in \bigotimes_{d=1}^{D} \left(\mathbb{R}^{m(i,d)} \otimes \mathbb{R}^{n(j,d)}\right)$, et les F-tuples $[\mathbf{u}]$ et $[\mathbf{b}]$ de tenseurs $\mathbf{u}_j \in \bigotimes_{d=1}^{D} \mathbb{R}^{n(j,d)}$ et $\mathbf{b}_i \in \bigotimes_{d=1}^{D} \mathbb{R}^{m(i,d)}$, pour $i, j = 1, \cdots, F$ respectivement, on définit le système d'équations linéaires suivant :

$$\boxed{[\![\mathbf{A}]\!] \overset{\mathrm{D}}{\cdot} [\mathbf{u}] = [\mathbf{b}] \quad \Leftrightarrow \quad \sum_{j=1}^{F} \mathbf{A}_{ij} \overset{\mathrm{D}}{\cdot} \mathbf{u}_j = \mathbf{b}_i\,, \quad \forall i = 1, \cdots, F}\,. \tag{A.13}$$

A.2.3 Transposée

- La transposée d'un tenseur $\mathbf{A} \in \bigotimes_{d=1}^{D} \left(\mathbb{R}^{m_d} \otimes \mathbb{R}^{n_d}\right)$ est le tenseur noté $\mathbf{A}' \in \bigotimes_{d=1}^{D} \left(\mathbb{R}^{n_d} \otimes \mathbb{R}^{m_d}\right)$ défini par :

$$\boxed{A'_{j_1 i_1 \cdots j_D i_D} = A_{i_1 j_1 \cdots i_D j_D}\,, \quad \begin{cases} \forall i_d = 1, \cdots, m_d \\ \forall j_d = 1, \cdots, n_d \\ \forall d = 1, \cdots, D \end{cases}}\,. \tag{A.14}$$

- La transposée d'un $F \times F$-tuple $[\![\mathbf{A}]\!]$ de tenseurs $\mathbf{A}_{ij} \in \bigotimes_{d=1}^{D} \left(\mathbb{R}^{m(i,d)} \otimes \mathbb{R}^{n(j,d)}\right)$ est le $F \times F$-tuple noté $[\![\mathbf{A}]\!]'$ défini par :

$$\boxed{\mathbf{A}'_{ji} = \mathbf{A}_{ij}\,, \quad \begin{cases} \forall i = 1, \cdots, F \\ \forall j = 1, \cdots, F \end{cases}}\,. \tag{A.15}$$

3. Un $F \times F$-tuple de tenseur d'ordre D est un tableau à deux dimensions de taille $F \times F$ et dont chaque case contient un tenseur d'ordre D.

A.2.4 Inverse

- L'inverse d'un tenseur $\mathbf{A} \in \bigotimes_{d=1}^{D} \left(\mathbb{R}^{n_d} \otimes \mathbb{R}^{n_d}\right)$ est le tenseur noté $\mathbf{A}^{-1} \in \bigotimes_{d=1}^{D} \left(\mathbb{R}^{n_d} \otimes \mathbb{R}^{n_d}\right)$ défini par :

$$\boxed{\mathbf{A}^{-1} \overset{\mathrm{D}}{\cdot} \mathbf{A} = \mathbf{I}}, \tag{A.16}$$

où $\mathbf{I}$ est le tenseur identité sur $\bigotimes_{d=1}^{D} \left(\mathbb{R}^{n_d} \otimes \mathbb{R}^{n_d}\right)$ défini par

$$\mathbf{I} = \bigotimes_{d=1}^{D} \mathbf{I}^d, \tag{A.17}$$

avec $\mathbf{I}^d$ la matrice identité sur $\mathbb{R}^{n_d} \otimes \mathbb{R}^{n_d}$ pour $d = 1, \cdots, D$, et l'opération $\mathbf{A} \overset{\mathrm{D}}{\cdot} \mathbf{B}$ est définie par

$$\mathbf{A} \overset{\mathrm{D}}{\cdot} \mathbf{B} = \mathbf{C} \quad \Leftrightarrow \quad \sum_{k_1} \cdots \sum_{k_D} A_{i_1 k_1 \cdots i_D k_D} B_{k_1 j_1 \cdots k_D j_D} = C_{i_1 j_1 \cdots i_D j_D}, \quad \forall i_d, \forall j_d, \forall d. \tag{A.18}$$

- L'inverse d'un $F \times F$-tuple $[\![\mathbf{A}]\!]$ de tenseurs $\mathbf{A}_{ij} \in \bigotimes_{d=1}^{D} \left(\mathbb{R}^{n(i,d)} \otimes \mathbb{R}^{n(j,d)}\right)$ est le $F \times F$-tuple noté $[\![\mathbf{A}]\!]^{-1}$ défini par :

$$\boxed{[\![\mathbf{A}]\!]^{-1} \overset{\mathrm{D}}{\cdot}. [\![\mathbf{A}]\!] = [\![\mathbf{I}]\!]}, \tag{A.19}$$

où $[\![\mathbf{I}]\!]$ est $F \times F$-tuple identité défini par

$$\mathbf{I}_{ij} = \begin{cases} \mathbf{I}_i & \text{pour } i = j \\ \mathbf{0}_{ij} & \text{pour } i \neq j \end{cases} \text{pour } i, j = 1, \cdots, F \tag{A.20}$$

avec $\mathbf{I}_i$ est le tenseur identité sur $\bigotimes_{d=1}^{D} \left(\mathbb{R}^{n(i,d)} \otimes \mathbb{R}^{n(i,d)}\right)$ et $\mathbf{0}_{ij}$ est le tenseur nul sur $\bigotimes_{d=1}^{D} \left(\mathbb{R}^{n(i,d)} \otimes \mathbb{R}^{n(j,d)}\right)$, et l'opération $[\![\mathbf{A}]\!] \overset{\mathrm{D}}{\cdot}. [\![\mathbf{B}]\!]$ est définie par

$$[\![\mathbf{A}]\!] \overset{\mathrm{D}}{\cdot}. [\![\mathbf{B}]\!] = [\![\mathbf{C}]\!] \quad \Leftrightarrow \quad \sum_{k} \mathbf{A}_{ik} \overset{\mathrm{D}}{\cdot} \mathbf{B}_{kj} = \mathbf{C}_{ij}, \quad \forall i, \forall j. \tag{A.21}$$

Bibliographie

[Abedi *et al.*, 2006] ABEDI, R., PETRACOVICI, B. et HABER, H. (2006). A space-time discontinuous galerkin method for linearized elastodynamics with element-wise momentum balance. *Comput. Methods Appl. Mech. Engrg.*, 195:3247–3273.

[Acar *et al.*, 2011] ACAR, E., DUNLAVY, D. et KOLDA, T. (2011). A scalable optimization approach for fitting canonical tensor decompositions. *Journal of Chemometrics*, 25: 67–86.

[Achenbach, 1973] ACHENBACH, J. (1973). *Wave propagation in elastic solids*. Elsevier.

[Aharoni et Bar-Yoseph, 1992] AHARONI, D. et BAR-YOSEPH, P. (1992). Mixed finite element formulations in the time domain for solution of dynamic problems. *Computational mechanics*, 9(5):359–374.

[Allaire, 2005] ALLAIRE, G. (2005). *Analyse numérique et optimisation : une introduction à la modélisation mathématique et à la simulation numérique*. Editions Ecole Polytechnique.

[Almeida, 2013] ALMEIDA, J. (2013). A basis for bounding the errors of proper generalised decomposition solutions in solid mechanics. *International Journal for Numerical Methods in Engineering*, 94:961–984.

[Ammar, 2010] AMMAR, A. (2010). The proper generalized decomposition : a powerful tool for model reduction. *International Journal of Material Forming*, 3:89–102.

[Ammar *et al.*, 2011] AMMAR, A., CHINESTA, F. et CUETO, E. (2011). Coupling finite elements and proper generalized decompositions. *International Journal for Multiscale Computational Engineering*, 9(1).

[Ammar *et al.*, 2012] AMMAR, A., CHINESTA, F., CUETO, E. et DOBLARÉ, M. (2012). Proper generalized decomposition of time-multiscale models. *International Journal for Numerical Methods in Engineering*, 90:569–596.

[Ammar *et al.*, 2010a] AMMAR, A., CHINESTA, F., DIEZ, P. et HUERTA, A. (2010a). An error estimator for separated representations of highly multidimensional models. *Computer Methods in Applied Mechanics and Engineering*, 199:1872–1880.

[Ammar *et al.*, 2010b] AMMAR, A., CHINESTA, F. et FALCÓ, A. (2010b). On the convergence of a greedy rank-one update algorithm for a class of linear systems. *Archives of Computational Methods in Engineering*, 17:473–486.

[Ammar *et al.*, 2013a] AMMAR, A., CUETO, A. et CHINESTA, F. (2013a). Nonincremental proper generalized decomposition solution of parametric uncoupled models defined in evolving domains. *International Journal for Numerical Methods in Engineering*, 93:887–904.

[Ammar *et al.*, 2013b] AMMAR, A., HUERTA, A., CHINESTA, F., CUETO, E. et LEYGUE, A. (2013b). Parametric solutions involving geometry : a step towards efficient shape optimization. *Computer Methods in Applied Mechanics and Engineering*.

[Ammar *et al.*, 2006] AMMAR, A., MOKDAD, B., CHINESTA, F. et KEUNINGS, R. (2006). A new family of solvers for some classes of multidimensional partial differential equations encountered in kinetic theory. *J. Non-Newtonian Fluid Mech.*, 139:153–176.

[Ammar *et al.*, 2007] AMMAR, A., MOKDAD, B., CHINESTA, F. et KEUNINGS, R. (2007). 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*, 144:98–121.

[Anderson *et al.*, 1999] ANDERSON, E., BAI, Z., BISCHOF, C., BLACKFORD, S., DEMMEL, J., DONGARRA, J., DU CROZ, J., GREENBAUM, A., HAMMARLING, S., MCKENNEY, A. et SORENSEN, D. (1999). *LAPACK User's Guide*. SIAM, (http ://www.netlib.org/lapack/lug/lapack_lug.html).

[Antoulas, 2005] ANTOULAS, A. (2005). *Approximation of large-scale dynamical systems*. SIAM.

[Aubry *et al.*, 1999] AUBRY, D., LUCAS, D. et TIE, B. (1999). Adaptive strategy for transient/coupled problems applications to thermoelasticity and elastodynamics. *Computer methods in applied mechanics and engineering*, 176(1):41–50.

[Bader *et al.*, 2012] BADER, B., KOLDA, T. *et al.* (2012). *MATLAB Tensor Toolbox Version 2.5*. http ://www.sandia.gov/ tgkolda/TensorToolbox.

[Ballani et Grasedyck, 2013] BALLANI, J. et GRASEDYCK, L. (2013). A projection method to solve linear system in tensor format. *Numer. Linear Algebra Appl.*, 20:27–43.

[Bamer et Bucher, 2012] BAMER, F. et BUCHER, C. (2012). Application of the proper orthogonal decomposition for linear and nonlinear structures under transient excitations. *Acta Mechanica*, 223:2549–2563.

[Barbarulo, 2012] BARBARULO, A. (2012). *On a PGD model order reduction technique for mid-frequency acoustic*. Thèse de doctorat, École normale supérieure de Cachan.

[Beringhier *et al.*, 2010] BERINGHIER, M., GUEGUEN, M. et GRANDIDIER, J. (2010). Solution of strongly coupled multiphysics problems using space-time separated representations. application to thermoviscoelasticity. *Archives of Computational Methods in Engineering*, 17:393–401.

[Berkooz *et al.*, 1993] BERKOOZ, G., HOLMES, P. et LUMLEY, J. (1993). The proper orthogonal decomposition in the analysis of turbulent flows. *Annual review of fluid mechanics*, 25:539–575.

[Beylkin et Mohlenkamp, 2002] BEYLKIN, G. et MOHLENKAMP, M. (2002). Numerical operator calculus in higher dimensions. *Proceedings of the National Academy of Sciences*, 99:10246–10251.

[Beylkin et Mohlenkamp, 2005] BEYLKIN, G. et MOHLENKAMP, M. (2005). Algorithms for numerical analysis in high dimensions. *SIAM J. Sci. Comput.*, 26:2133–2159.

[Billaud-Friess *et al.*, 2013] BILLAUD-FRIESS, M., NOUY, A. et ZAHM, O. (2013). A tensor approximation method based on ideal minimal residual formulations for the solution of high dimensional problems, arxiv :1304.6126.

[Bognet *et al.*, 2012] BOGNET, B., BORDEU, F., CHINESTA, F., LEYGUE, A. et POITOU, A. (2012). Advanced simulation of models defined in plate geometries : 3d solutions with 2d computational complexity. *Computer Methods in Applied Mechanics and Engineering*, 201:1–12.

[Bonithon *et al.*, 2011] BONITHON, G., JOYOT, P., CHINESTA, F. et VILLON, P. (2011). Non-incremental boundary element discretization of parabolic models based on the use of the proper generalized decompositions. *Engineering Analysis with Boundary Elements*, 35:2–17.

[Bordeu, 2013] BORDEU, F. (2013). *PXDMF : A File Format for Separated Variables Problems - Version 1.4*. http ://rom.research-centrale-nantes.com.

[Boucinha *et al.*, 2013a] BOUCINHA, L., AMMAR, A., GRAVOUIL, A. et NOUY, A. (2013a). Ideal minimal residual-based proper generalized decomposition for non-symmetric multi-field models – application to transient elastodynamics in space-time domain (accepted). *Computer Methods in Applied Mechanics and Engineering*.

[Boucinha *et al.*, 2013b] BOUCINHA, L., GRAVOUIL, A. et AMMAR, A. (2013b). Space-time proper generalized decompositions for the resolution of transient elastodynamic models. *Computer Methods in Applied Mechanics and Engineering*, 255:67–88.

[Cancès *et al.*, 2011] CANCÈS, E., EHRLACHER, V. et LELIEVRE, T. (2011). Convergence of a greedy algorithm for high-dimensional convex nonlinear problems. *Mathematical Models and Methods in Applied Sciences*, 21:2433–2467.

[Cancès *et al.*, 2012] CANCÈS, E., EHRLACHER, V. et LELIEVRE, T. (2012). Greedy algorithms for high-dimensional non-symmetric linear problems. *arXiv preprint arXiv :1210.6688*.

[Cannarozzi et Mancuso, 1995] CANNAROZZI, M. et MANCUSO, M. (1995). Formulation and analysis of variationnal methods for time integration of linear elastodynamics. *Comput. Methods Appl. Mech. Eng.*, 127:241–257.

[Carroll et Chang, 1970] CARROLL, J. et CHANG, J. (1970). Analysis of individual differences in multidimensional scaling via an n-way generalization of eckart-young decomposition. *Psychometrika*, 35:283–319.

[Cavin *et al.*, 2005] CAVIN, P., GRAVOUIL, A., LUBRECHT, A. et COMBESCURE, A. (2005). Automatic energy conserving space-time refinement for linear dynamic structural problems. *Int. J. Numer. Meth. Engng.*, 64:304–321.

[Chevreuil et Nouy, 2012] CHEVREUIL, M. et NOUY, A. (2012). Model order reduction based on proper generalized decomposition for the propagation of uncertainties in structural dynamics. *International Journal for Numerical Methods in Engineering*, 89:241–268.

[Chinesta *et al.*, 2010] CHINESTA, F., AMMAR, A. et CUETO, E. (2010). Recent advances and new challenges in the use of the proper generalized decomposition for solving multidimensional models. *Archives of Computational methods in Engineering*, 17: 327–350.

[Chinesta *et al.*, 2008] CHINESTA, F., AMMAR, A., LEMARCHAND, F., BEAUCHENE, P. et BOUST, F. (2008). Alleviating mesh constraints : model reduction, parallel time integration and high resolution homogenization. *Comput. Methods Appl. Mech. Engrg.*, 197:400–413.

[Chinesta *et al.*, 2011] CHINESTA, F., LADEVÈZE, P. et CUETO, E. (2011). A short review on model order reduction based on proper generalized decomposition. *Arch. Comput Methods Eng*, 404:395–404.

[Chinesta *et al.*, 2013] CHINESTA, F., LEYGUE, A., BORDEU, F., AGUADO, J., CUETO, E., GONZALEZ, D., ALFARO, I., AMMAR, A. et HUERTA, A. (2013). Pgd-based computational vademecum for efficient design, optimization and control. *Archives of Computational Methods in Engineering*, 20:31–59.

[Cohen *et al.*, 2012] COHEN, A., DAHMEN, W. et WELPER, G. (2012). Adaptivity and variational stabilization for convection-diffusion equations. *ESAIM : Mathematical Modelling and Numerical Analysis*, 46:1247–1273.

[Dahmen *et al.*, 2012] DAHMEN, W., HUANG, C., SCHWAB, C. et WELPER, G. (2012). Adaptive petrov-galerkin methods for first order transport equations. *SIAM Journal on Numerical Analysis*, 50:2420–2445.

[De Silva et Lim, 2008] DE SILVA, V. et LIM, L. (2008). Tensor rank and the ill-posedness of the best low-rank approximation problem. *SIAM Journal on Matrix Analysis and Applications*, 30:1084–1127.

[De Sterck, 2012] DE STERCK, H. (2012). A nonlinear gmres optimization algorithm for canonical tensor decomposition. *SIAM Journal on Scientific Computing*, 34:A1351–A1379.

[De Sterck et Miller, 2013] DE STERCK, H. et MILLER, K. (2013). An adaptive algebraic multigrid algorithm for low-rank canonical tensor decomposition. *SIAM Journal on Scientific Computing*, 35:B1–B24.

[Ding et Chen, 2005] DING, F. et CHEN, T. (2005). Gradient based iterative algorithms for solving a class of matrix equations. *Automatic Control, IEEE Transactions on*, 50(8):1216–1221.

[Ding et Chen, 2006] DING, F. et CHEN, T. (2006). On iterative solutions of general coupled matrix equations. *SIAM Journal on Control and Optimization*, 44(6):2269–2284.

[Dumon *et al.*, 2011] DUMON, A., ALLERY, C. et AMMAR, A. (2011). Proper general decomposition (pgd) for the resolution of navier-stokes equations. *Journal of Computational Physics*, 230:1387–1407.

[Dumon *et al.*, 2013] DUMON, A., ALLERY, C. et AMMAR, A. (2013). Proper generalized decomposition method for incompressible navier–stokes equations with a spectral discretization. *Applied Mathematics and Computation*, 219:8145–8162.

[Dureisseix *et al.*, 2003] DUREISSEIX, D., LADEVÈZE, P. et SCHREFLER, B. (2003). A latin computational strategy for multiphysics problems : application to poroelasticity. *International Journal for Numerical Methods in Engineering*, 56:1489–1510.

[Eckart et Young, 1936] ECKART, C. et YOUNG, G. (1936). The approximation of one matrix by another of lower rank. *Psychometrika*, 1:211–218.

[Eftekhar Azam et Mariani, 2013] EFTEKHAR AZAM, S. et MARIANI, S. (2013). Investigation of computational and accuracy issues in pod-based reduced order modeling of dynamic structural systems. *Engineering Structures*, 54:150–167.

[Espig *et al.*, 2012] ESPIG, M., HACKBUSCH, W., ROHWEDDER, T. et SCHNEIDER, R. (2012). Variational calculus with sums of elementary tensors of fixed rank. *Numerische Mathematik*, 122:469–488.

[Falco et Nouy, 2011] FALCO, A. et NOUY, A. (2011). A proper generalized decomposition for the solution of elliptic problems in abstract form by using a functional eckart-young approach. *Journal of Mathematical Analysis and applications*, 376: 469–480.

[Falcó et Nouy, 2012] FALCÓ, A. et NOUY, A. (2012). Proper generalized decomposition for nonlinear convex problems in tensor banach spaces. *Numerische Mathematik*, 121:503–530.

[Feeny et Kappagantu, 1998] FEENY, B. et KAPPAGANTU, R. (1998). On the physical interpretation of proper orthogonal modes in vibrations. *Journal of Sound and Vibration*, 211(4):607–616.

[Felippa *et al.*, 2001] FELIPPA, C., PARK, K. et FARHAT, C. (2001). Partitioned analysis of coupled mechanical systems. *Computer methods in applied mechanics and engineering*, 190:3247–3270.

[French et Peterson, 1996] FRENCH, D. et PETERSON, T. (1996). A continuous space-time finite element method for the wave equation. *Mathematics of Computation of the American Mathematical Society*, 65:491–506.

[Friedman et Kline, 1955] FRIEDMAN, B. et KLINE, M. (1955). An abstract formulation of the method of separation of variables. Rapport technique, New York University, Institute of Mathematical Sciences.

[Géradin et Rixen, 1997] GÉRADIN, M. et RIXEN, D. (1997). *Mechanical vibrations : theory and application to structural dynamics*. Wiley.

[Giner *et al.*, 2013] GINER, E., BOGNET, B., RÒDENAS, J., LEYGUE, A., FUENMAYOR, F. et F, C. (2013). The proper generalized decomposition (pgd) as a numerical procedure

to solve 3d cracked plates in linear elastic fracture mechanics. *International Journal of Solids and Structures*, 50:1710–1720.

[Giorgiani *et al.*, 2013] GIORGIANI, G., MODESTO, D., FERNÁNDEZ-MÉNDEZ, S. et HUERTA, A. (2013). High-order continuous and discontinuous galerkin methods for wave problems. *International Journal for Numerical Methods in Fluids*.

[Giraldi, 2012] GIRALDI, L. (2012). *Contributions aux méthodes de calcul basées sur l'approximation de tenseurs et applications en mécanique numérique*. Thèse de doctorat, École Centrale de Nantes.

[Giraldi *et al.*, 2013] GIRALDI, L., NOUY, A. et LEGRAIN, G. (2013). Low-rank approximate inverse for preconditioning tensor-structured linear systems. *arXiv preprint arXiv :1304.6004*.

[Glösmann et Kreuzer, 2009] GLÖSMANN, P. et KREUZER, E. (2009). On the application of karhunen–loève transform to transient dynamic systems. *Journal of Sound and Vibration*, 328:507–519.

[González *et al.*, 2010] GONZÁLEZ, D., AMMAR, A., CHINESTA, F. et CUETO, E. (2010). Recent advances on the use of separated representations. *International Journal for Numerical Methods in Engineering*, 81:637–659.

[Grasedyck *et al.*, 2013] GRASEDYCK, L., KRESSNER, D. et TOBLER, C. (2013). A literature survey of low-rank tensor approximation techniques. *arXiv preprint arXiv :1302.7121*.

[Gravouil, 2000] GRAVOUIL, A. (2000). *Méthode multi-échelles en temps et en espace avec décomposition de domaines pour la dynamique non-linéaire des structures*. Thèse de doctorat, École normale supérieure de Cachan.

[Hackbusch, 2003] HACKBUSCH, W. (2003). Multigrid methods for fem and bem applications. *Encyclopedia of Computational Mechanics*.

[Hackbusch, 2012] HACKBUSCH, W. (2012). *Tensor Spaces and Numerical Tensor Calculus*. Springer.

[Hackbusch et Kühn, 2009] HACKBUSCH, W. et KÜHN, S. (2009). A new scheme for the tensor representation. *Journal of Fourier Analysis and Applications*, 15:706–722.

[Ham et Bathe, 2012] HAM, S. et BATHE, K. (2012). A finite element method enriched for wave propagation problems. *Computers & Structures*, 94:1–12.

[Harshman, 1970] HARSHMAN, R. (1970). Foundations of the parafac procedure : models and conditions for an explanatory multimodal factor analysis.

[Hilber et Hughes, 1978] HILBER, H. M. et HUGHES, T. J. (1978). Collocation, dissipation and overshoot for time integration schemes in structural dynamics. *Earthquake Engineering & Structural Dynamics*, 6:99–117.

[Hübner *et al.*, 2004] HÜBNER, B., WALHORN, E. et DINKLER, D. (2004). A monolithic approach to fluid–structure interaction using space–time finite elements. *Computer methods in applied mechanics and engineering*, 193:2087–2104.

[Huerta *et al.*, 2013] HUERTA, A., ANGELOSKI, A., ROCA, X. et PERAIRE, J. (2013). Efficiency of high-order elements for continuous and discontinuous galerkin methods. *Int. J. Numer. Meth. Engng.*, doi : 10.1002/nme.4547.

[Hughes, 1987] HUGHES, T. (1987). *The finite element method : linear static and dynamic finite element analysis*. Dover Publications.

[Hughes et Hulbert, 1988] HUGHES, T. et HULBERT, G. (1988). Space-time finite element methods for elastodynamics : formulations and error estimates. *Computer methods in applied mechanics and engineering*, 66(3):339–363.

[Hughes *et al.*, 2008] HUGHES, T. J., REALI, A. et SANGALLI, G. (2008). Duality and unified analysis of discrete approximations in structural dynamics and wave propagation : Comparison of p-method finite elements with k-method nurbs. *Computer methods in applied mechanics and engineering*, 197:4104–4124.

[Hulbert, 1992] HULBERT, G. (1992). Time finite element methods for structural dynamics. *International Journal for Numerical Methods in Engineering*, 33:307–331.

[Hulbert, 2004] HULBERT, G. (2004). Computational structural dynamics. *Encyclopedia of Computational Mechanics*.

[Hulbert et Hughes, 1990] HULBERT, G. M. et HUGHES, T. J. (1990). Space-time finite element methods for second-order hyperbolic equations. *Computer Methods in Applied Mechanics and Engineering*, 84:327–348.

[Idesman, 2007] IDESMAN, A. (2007). A new high-order accurate continuous galerkin method for linear elastodynamics problems. *Computational Mechanics*, 40(2):261–279.

[Ihlenburg et Babuška, 1995] IHLENBURG, F. et BABUŠKA, I. (1995). Finite element solution of the helmholtz equation with high wave number part i : The h-version of the fem. *Computers & Mathematics with Applications*, 30:9–37.

[Ihlenburg et Babuška, 1997] IHLENBURG, F. et BABUŠKA, I. (1997). Finite element solution of the helmholtz equation with high wave number part ii : The hp version of the fem. *SIAM Journal on Numerical Analysis*, 34:315–358.

[Kerschen et Golinval, 2002] KERSCHEN, G. et GOLINVAL, J. (2002). Physical interpretation of the proper orthogonal modes using the singular value decomposition. *Journal of Sound and Vibration*, 249(5):849–865.

[Kerschen *et al.*, 2005] KERSCHEN, G., GOLINVAL, J., VAKAKIS, A. et BERGMAN, L. (2005). The method of proper orthogonal decomposition for dynamical characterization and order reduction of mechanical systems : an overview. *Nonlinear Dynamics*, 41: 147–169.

[Khoromskij et Schwab, 2011] KHOROMSKIJ, B. et SCHWAB, C. (2011). Tensor-structured galerkin approximation of parametric and stochastic elliptic pdes. *SIAM Journal on Scientific Computing*, 33:364–385.

[Kolda et Bader, 2009] KOLDA, T. et BADER, B. (2009). Tensor decompositions and applications. *SIAM review*, 51:455–500.

[Kressner et Tobler, 2010] KRESSNER, D. et TOBLER, C. (2010). Krylov subspace methods for linear systems with tensor product structure. *SIAM journal on matrix analysis and applications*, 31:1688–1714.

[Kunthong et Thompson, 2005] KUNTHONG, P. et THOMPSON, L. (2005). An efficient solver for the high-order accurate time-discontinuous galerkin (tdg) method for second-order hyperbolic systems. *Finite elements in analysis and design*, 41(7):729–762.

[Ladevèze, 1999] LADEVÈZE, P. (1999). *Nonlinear Computational Structural Mechanics - New Approaches and Non-Incremental Methods of Calculation.* Springer-Verlag.

[Ladevèze et Chamoin, 2011] LADEVÈZE, P. et CHAMOIN, L. (2011). On the verification of model reduction methods based on the proper generalized decomposition. *Computer Methods in Applied Mechanics and Engineering*, 200:2032–2047.

[Ladevèze et Nouy, 2003] LADEVÈZE, P. et NOUY, A. (2003). On a multiscale computational strategy with time and space homogenization for structural mechanics. *Computer Methods in Applied Mechanics and Engineering*, 192:3061–3087.

[Ladevèze *et al.*, 2010] LADEVÈZE, P., PASSIEUX, J. et NÉRON, D. (2010). The latin multiscale computational method and the proper generalized decomposition. *Computer Methods in Applied Mechanics and Engineering*, 199:1287–1296.

[Lassila *et al.*, 2013] LASSILA, T., MANZONI, A., QUARTERONI, A. et ROZZA, G. (2013). Model order reduction in fluid dynamics : challenges and perspectives. Rapport technique 22.2013, MATHICSE Technical Report.

[Le Bris *et al.*, 2009] LE BRIS, C., LELIEVRE, T. et MADAY, Y. (2009). Results and questions on a nonlinear approximation approach for solving high-dimensional partial differential equations. *Constructive Approximation*, 30:621–651.

[Leygue et Verron, 2010] LEYGUE, A. et VERRON, E. (2010). A first step towards the use of proper general decomposition method for structural optimization. *Archives of Computational Methods in Engineering*, 17:465–472.

[Li et Wiberg, 1996] LI, X. et WIBERG, N. (1996). Structural dynamic analysis by a time-discontinuous galerkin finite element method. *International journal for numerical methods in engineering*, 39(12):2131–2152.

[Li et Wiberg, 1998] LI, X. et WIBERG, N. (1998). Implementation and adaptivity of a space-time finite element method for structural dynamics. *Computer methods in applied mechanics and engineering*, 156(1):211–229.

[Liang *et al.*, 2002] LIANG, Y., LEE, H., LIM, S., LIN, W., LEE, K. et WU, C. (2002). Proper orthogonal decomposition and its applications - part i : Theory. *Journal of Sound and Vibration*, 252:527–544.

[Lynch *et al.*, 1964] LYNCH, R., RICE, J. et THOMAS, D. (1964). Direct solution of partial difference equations by tensor product methods. *Numerische Mathematik*, 6:185–199.

[Mahjoubi *et al.*, 2011] MAHJOUBI, N., GRAVOUIL, A., COMBESCURE, A. et GREFFET, N. (2011). A monolithic energy conserving method to couple heterogeneous time integrators with incompatible time steps in structural dynamics. *Computer Methods in Applied Mechanics and Engineering*, 200:1069–1086.

[Matthies et Zander, 2012] MATTHIES, H. et ZANDER, E. (2012). Solving stochastic systems with low-rank tensor compression. *Linear Algebra and its Applications*, 436:3819–3838.

[Michler *et al.*, 2004] MICHLER, C., HULSHOFF, S., VAN BRUMMELEN, E. et DE BORST, R. (2004). A monolithic approach to fluid-structure interaction. *Computers & fluids*, 33:839–848.

[Modesto *et al.*, 2012] MODESTO, D., GIORGIANI, G., ZLOTNIK, S. et HUERTA, A. (2012). Efficiency and accuracy of high-order computations and reduced order modelling in coastal engineering wave propagation problems. *In Proceedings of the ECCOMAS Young Investigators Conferences 2012*.

[Néron et Dureisseix, 2008] NÉRON, D. et DUREISSEIX, D. (2008). A computational strategy for thermo-poroelastic structures with a time-space interface coupling. *International Journal for Numerical Methods in Engineering*, 75:1053–1084.

[Néron, 2010] NÉRON, Dand Ladevèze, P. (2010). Proper generalized decomposition for multiscale and multiphysics problems. *Archives of Computational Methods in Engineering*, 17:351–372.

[Newmark, 1959] NEWMARK, N. (1959). A method of computation for structural dynamics. *In Proc. ASCE*, volume 85, pages 67–94.

[Niroomandi *et al.*, 2012] NIROOMANDI, S., ALFARO, I., GONZALEZ, D., CUETO, E. et CHINESTA, F. (2012). Real-time simulation of surgery by reduced-order modeling and x-fem techniques. *International Journal for Numerical Methods in Biomedical Engineering*, 28:574–588.

[Nouy, 2007] NOUY, A. (2007). A generalized spectral decomposition technique to solve a class of linear stochastic partial differential equations. *Comput. Methods Appl. Mech. Engrg.*, 196:4521–4537.

[Nouy, 2008] NOUY, A. (2008). Generalized spectral decomposition method for solving stochastic finite element equations : invariant subspace problem and dedicated algorithms. *Comput. Methods Appl. Mech. Engrg.*, 197:4718–4736.

[Nouy, 2010a] NOUY, A. (2010a). A priori model reduction through proper generalized decomposition for solving time-dependent partial differential equations. *Comput. Methods Appl. Mech. Engrg.*, 199:1603–1626.

[Nouy, 2010b] NOUY, A. (2010b). Proper generalized decompositions and separated representations for the numerical solution of high dimensional stochastic problems. *Arch. Comput Methods Eng*, 17:403–434.

[Nouy *et al.*, 2011] NOUY, A., CHEVREUIL, M. et SAFATLY, E. (2011). Fictitious domain method and separated representations for the solution of boundary value problems

on uncertain parameterized domains. *Computer Methods in Applied Mechanics and Engineering*, 200:3066–3082.

[Nouy et Ladevèze, 2004] NOUY, A. et LADEVÈZE, P. (2004). Multiscale computational strategy with time and space homogenization : a radial-type approximation technique for solving micro problems. *Int. J. Multiscale Comput. Engrg.*, 170:557–574.

[Ohayon et Soize, 1998] OHAYON, R. et SOIZE, C. (1998). *Structural acoustics and vibration : Mechanical models, variational formulations and discretization*. Academic Press.

[Passieux *et al.*, 2010] PASSIEUX, J.-C., LADEVÈZE, P. et NÉRON, D. (2010). A scalable time-space multiscale domain decomposition method : adaptive time scale separation. *Computational Mechanics*, 46:621–633.

[Placzek *et al.*, 2008] PLACZEK, A., TRAN, D. et OHAYON, R. (2008). Hybrid proper orthogonal decomposition formulation for linear structural dynamics. *Journal of Sound and Vibration*, 318:943–964.

[Prenter, 1975] PRENTER, P. (1975). *Splines and variational methods*. John Wiley & Sons.

[Qu, 2004] QU, Z. (2004). *Model order reduction techniques : with applications in finite element analysis*. Springer.

[Rethore *et al.*, 2005] RETHORE, J., GRAVOUIL, A. et COMBESCURE, A. (2005). A combined space-time extended finite element method. *International Journal for Numerical Methods in Engineering*, 64(2):260–284.

[Ryckelynck, 2005] RYCKELYNCK, D. (2005). A priori hyperreduction method : an adaptive approach. *Journal of Computational Physics*, 202:346–366.

[Ryckelynck *et al.*, 2006] RYCKELYNCK, D., CHINESTA, F., CUETO, E. et AMMAR, A. (2006). On the a priori model reduction : Overview and recent developments. *Archives of Computational Methods in Engineering*, 13:91–128.

[Schmidt, 1907] SCHMIDT, E. (1907). Zur theorie der linearen und nichtlinearen integralgleichungen. i teil. entwicklung willkurlichen funktionen nach system vorgeschriebener. *Mathematische Annalen*, 63:433–476.

[Tamellini *et al.*, 2012] TAMELLINI, L., LE MAITRE, O. et NOUY, A. (2012). Model reduction based on proper generalized decomposition for the stochastic steady incompressible navier-stokes equations.

[Van Loan, 2000] VAN LOAN, C. (2000). The ubiquitous kronecker product. *Journal of Computational and Applied Mathematics*, 123:85–100.

[Van Loan et Pitsianis, 1992] VAN LOAN, C. et PITSIANIS, N. (1992). Approximation with kronecker products. Rapport technique, Cornell University.

[Volkwein, 2008] VOLKWEIN, S. (2008). Model reduction using proper orthogonal decomposition. *Lecture Notes, Institute of Mathematics and Scientific Computing, University of Graz. see http ://www. uni-graz. at/imawww/volkwein/POD.pdf*.

[Zauderer, 1989] ZAUDERER, E. (1989). *Partial differential equations of applied mathematics*. Wiley.

[Zhou *et al.*, 2009] ZHOU, B., DUAN, G. et LI, Z. (2009). Gradient based iterative algorithm for solving coupled matrix equations. *Systems & Control Letters*, 58:327–333.

FOLIO ADMINISTRATIF

THÈSE SOUTENUE DEVANT L'INSTITUT NATIONAL DES SCIENCES APPLIQUÉES DE LYON

NOM : BOUCINHA DATE de SOUTENANCE :15 novembre 2013

Prénoms : Lucas

TITRE : Réduction de modèle a priori par séparation de variables espace-temps – Application en dynamique transitoire

NATURE : Doctorat Numéro d'ordre : 2013-ISAL-XXX

École doctorale : MEGA

Spécialité : Mécanique - Génie Mécanique - Génie Civil

RÉSUMÉ :

La simulation numérique des phénomènes physiques est devenue un élément incontournable dans la boite à outils de l'ingénieur mécanicien. Des outils robustes et modulables, basés sur les méthodes classiques d'approximation, sont désormais couramment utilisés dans l'industrie. Cependant, ces outils nécessitent des moyens de calculs importants lorsqu'ils sont utilisés pour résoudre des problèmes complexes. Même si les progrès remarquables de l'industrie informatique rendent de tels moyens de calcul toujours plus abordables, il s'avère aujourd'hui nécessaire de proposer des méthodes d'approximation innovantes permettant de mieux exploiter les ressources informatiques disponibles. Les méthodes de réduction de modèle sont présentées comme un candidat idéal pour atteindre cet objectif. Parmi celles-ci, les méthodes basées sur la construction d'une approximation à variables séparées se sont révélées être très efficaces pour approcher la solution d'une grande variété de problèmes, réduisant les coûts numériques de plusieurs ordres de grandeur. Néanmoins, l'efficacité de ces méthodes dépend considérablement du problème traité. Dans ce manuscrit, on se propose d'évaluer l'intérêt d'une approximation à variables séparées espace-temps dans le cadre de problèmes académiques de dynamique transitoire. On définit tout d'abord la meilleure approximation (au sens d'un problème de minimisation) de la solution d'un problème transitoire, sous la forme d'une représentation à variables séparées espace-temps. Le calcul de cette approximation étant basé sur l'hypothèse que la solution du problème de référence est connue (méthode *a posteriori*), la suite du manuscrit est dédiée à la construction d'une telle approximation sans autres connaissances a priori sur la solution de référence, que les opérateurs du problème espace-temps dont elle est solution (méthode *a priori*). Un formalisme générique, basé sur une représentation tensorielle des opérateurs du problème espace-temps est alors introduit dans un cadre multichamps. On développe ensuite un solveur exploitant ce format générique, pour construire une approximation à variables séparées espace-temps de la solution d'un problème transitoire. Ce solveur est basé sur la décomposition généralisée propre de la solution (Proper Generalized Decomposition - PGD). Un état de l'art des algorithmes existants permet alors d'évaluer l'efficacité des définitions classiques de la PGD pour approcher la solution de problèmes académiques de dynamique transitoire. Les résultats obtenus mettant en défaut l'optimalité de la PGD la plus robuste, une nouvelle définition, récemment introduite dans la littérature, est appliquée dans un cadre multichamps à la résolution d'un problème d'élastodynamique 2D. Cette nouvelle définition, basée sur la minimisation du résidu dans une norme idéale, permet finalement d'obtenir une très bonne approximation de la meilleure approximation de rang donné, sans avoir à calculer un grand nombre de modes espace-temps.

MOTS-CLÉS : dynamique transitoire, séparation de variables espace-temps, POD/PGD

Laboratoire(s) de recherche : Laboratoire de Mécanique des Contacts et des Solides
UMR CNRS 5514 - INSA de Lyon
20, avenue Albert Einstein
69621 Villeurbanne Cedex FRANCE

Directeur de thèse : Anthony GRAVOUIL

Président du jury : ...

Composition du jury : Pierre LADEVÈZE Francisco CHINESTA
Antonio HUERTA Amine AMMAR
David RYCKELYNCK Anthony GRAVOUIL