

Réduction de modèles de problèmes paramétriques en mécanique non linéaire à l'aide de Code Aster et Mordicus

E. Agouzal^{1,2}, J.P. Argaud¹, M. Bergmann²,
G. Ferté¹, T. Taddei²

¹ Electricité de France (EDF) R&D, Saclay, France, {eki.agouzal, jean-philippe.argaud, guilhem.ferte}@edf.fr

² IMB, UMR 5251, Univ. Bordeaux; 33400, Talence, France. Inria Bordeaux Sud-Ouest, Team MEMPHIS; 33400, Talence, France, {michel.bergmann, tommaso.taddei}@inria.fr

Résumé — Nous proposons une approche générale de réduction de modèles (ROM) pour des modèles paramétriques en mécanique quasi-statique non-linéaire, basée sur une méthode POD, et une méthode de quadrature empirique. Le présent travail traite d'une implémentation non-intrusive interfacée avec le logiciel Code Aster et la librairie Python Mordicus. Nous présentons une analyse de la méthode avec un cas de validation en élasticité linéaire isotrope, en certifiant la qualité de l'approximation sur l'espace réduit à l'aide d'estimateurs corrélés à l'erreur d'approximation et économiques en temps de calcul.

Mots clés — réduction de modèles, études paramétriques, mécanique non-linéaire

1 Introduction

La simulation numérique est de plus en plus utilisée par les ingénieries d'EDF, très souvent dans un contexte de multiples évaluations d'une même étude dans des configurations légèrement différentes. Ces variantes peuvent consister en une différence de signaux d'entrée, de paramètres réels du modèle (*étude paramétrique*), voire de géométrie. Ces approches demandent de nombreuses évaluations du modèle numérique. Si le coût unitaire est important, alors le coût de calcul total devient vite prohibitif. La *réduction de modèles* est une famille d'algorithmes permettant de réduire drastiquement le coût marginal associé à une évaluation, en tirant parti de l'information des évaluations précédentes. Dans ce travail, nous présenterons l'implémentation peu intrusive d'une méthode générale à la mécanique quasi-statique avec comportement non linéaire.

2 Formulation du problème physique et éléments théoriques

2.1 Formulation variationnelle du problème mécanique

Nous étudions une classe générique de problèmes non-linéaires en mécanique. Nous notons x la variable d'espace d'un domaine suffisamment régulier $\Omega \subset \mathbb{R}^d$ ($d = 2$ ou 3), et $t \in [0, T]$ la variable temporelle. Nous considérons un vecteur de paramètres μ appartenant à un compact $\mathcal{P} \subset \mathbb{R}^p$. Nous notons \mathbf{u} la variable primale du problème (déplacement), et \mathcal{V} l'espace de Hilbert auquel il appartient. Nous nous restreignons à des cas où une force volumique est appliquée au système. Le problème paramétrique à l'étude est le suivant : étant donné $\mu \in \mathcal{P}$, trouver $\mathbf{u}(\mu) \in \mathcal{V}$:

$$\forall \mathbf{v} \in \mathcal{V}, \int_{\Omega} \underline{\underline{\sigma}}(\mathbf{u}(\mu), \mathcal{H}(\mathbf{u}(t'; \mu), t' \leq t)) : \underline{\underline{\varepsilon}}(\mathbf{v}) \, dx = \int_{\Omega} \mathbf{f} \cdot \mathbf{v} \, dx$$

avec des conditions initiales et aux bords adéquates, et où $\mathcal{H}(\mathbf{u}(t'; \mu), t' \leq t)$ représente l'histoire des déplacements jusqu'au temps actuel t . L'approximation de la solution par une méthode de type éléments finis s'obtient en substituant l'espace \mathcal{V} par le sous-espace éléments finis \mathcal{V}^{hf} , que nous appellerons espace haute-fidélité.

Nous nous restreignons à des problèmes quasi-statiques, où seule la connaissance du dernier pas de temps est suffisante dans la formulation variationnelle. Nous considérons une discrétisation temporelle en

K pas de temps $\{t_k\}_{k=1}^K$. Le problème haute-fidélité devient : étant donné $\mu \in \mathcal{P}$, trouver $\{\mathbf{u}(t_k; \mu)\}_{k=1}^K \in (\mathcal{V}^{\text{hf}})^K$:

$$\forall k \in \llbracket 1, K \rrbracket, \forall \mathbf{v} \in \mathcal{V}^{\text{hf}}, \int_{\Omega} \underline{\underline{\sigma}}(\mathbf{u}^k(\mu), \mathbf{u}^{k-1}(\mu)) : \underline{\underline{\varepsilon}}(\mathbf{v}) \, dx = \int_{\Omega} \mathbf{f} \cdot \mathbf{v} \, dx$$

En notant \mathcal{N} le nombre de noeuds de notre maillage, le problème se ramène à une formulation discrétisée : trouver les vecteurs $\{\mathbf{U}^k(\mu)\}_{k=1}^K \in (\mathbb{R}^{d\mathcal{N}})^K$ vérifiant :

$$\forall k \in \llbracket 1, K \rrbracket, \mathbf{R}(\mathbf{U}^k(\mu), \mathbf{U}^{k-1}(\mu)) = 0$$

2.2 Réduction de modèles par projection de Galerkin

Nous proposons une approche de réduction de modèles par projection de Galerkin [2]. La solution réduite $\mathbf{u}_N(\mu)$ est déterminée sur un sous-espace \mathcal{V}_N , appelé espace réduit. Nous introduisons la matrice \mathbb{Z}_u des vecteurs de la base réduite en déplacement $\forall j \in \llbracket 1, N_u \rrbracket \quad [\mathbb{Z}_u]_{:,j} = \zeta_{u,j}$. Nous cherchons alors des vecteurs solution sous la forme :

$$\mathbf{U}_N(t_k; \mu) = \sum_{j=1}^{N_u} \Lambda_u^j(t_k, \mu) \zeta_{u,j} = \mathbb{Z}_u \cdot [\Lambda_u^j(t_k, \mu)]_{j=1}^{N_u} \quad (1)$$

où le vecteur $[\Lambda_u^j(t_k, \mu)]_{j=1}^{N_u}$ contient les coordonnées réduites du déplacement au temps t_k . Une telle résolution revient à considérer la projection du résidu :

$$\forall k \in \llbracket 1, K \rrbracket, \mathbb{Z}_u^T \mathbf{R}(\mathbf{U}^k(\mu), \mathbf{U}^{k-1}(\mu)) = 0 \quad (2)$$

La non-linéarité de l'opérateur \mathbf{R} implique que cette opération ne diminuera pas drastiquement le coût de calcul. Pour résoudre cette problématique, une loi de quadrature empirique $\underline{\rho}^{\text{eq}}$ est construite (voire section 3.1.2).

Bien plus, la connaissance de l'état mécanique du système nécessite de disposer des contraintes dans le matériau. Les contraintes sont elles aussi décomposées sous une forme similaire à (1), c'est-à-dire sur une base réduite $\text{span}\{\zeta_{\sigma,j}\}_{j=1}^{N_{\sigma}}$, avec les coordonnées $\{\Lambda_{\sigma}^j(t_k, \mu)\}_{j=1}^{N_{\sigma}}$. Cette base sera utilisée pour définir un indicateur d'erreur *a posteriori* (section 3.2.2).

3 Méthodologie de l'algorithme POD-Greedy

3.1 Problème sans variabilité paramétrique (*Solution Reproduction Problem*)

Nous omettons dans un premier temps la variabilité en μ . Nous présentons ici la stratégie pour la construction d'une base réduite et d'un maillage réduit par quadrature empirique pour un paramètre donné. Notre objectif est ici de reproduire le résultat obtenu lors d'une simulation haute-fidélité par le biais de notre problème réduit. Cette approche se décompose en deux temps : une phase *offline* où nous construisons une base réduite (section 3.1.1), et un maillage réduit (section 3.1.2), puis une phase *online*, qui correspond aux calculs des coefficients des modes pour les déplacements et les contraintes. Le calcul des coefficients pour les contraintes nécessitent une étape supplémentaire avec une procédure de Gappy-POD (section 3.1.3).

3.1.1 Compression des données

Nous avons recours à la méthode des *snapshots* ([11]) pour générer la base réduite \mathbb{Z}_u . Nous définissons la matrice de Gram $\mathbb{C} \in \mathbb{R}^{K \times K}$ associée à un produit scalaire choisi $\mathbb{C}_{i,j} = (\mathbf{U}^{\text{hf},i}(\mu), \mathbf{U}^{\text{hf},j}(\mu))_*$. Le problème aux valeurs propres est alors résolu par une procédure de SVD tronquée, de manière à disposer des couples de valeurs propres et de vecteurs propres $(\lambda_q, \Psi_q)_{q=1}^{N_u}$ associés à la matrice \mathbb{C} . Le choix du nombre de valeurs propres conservé se fonde sur le critère en énergie suivant :

Algorithm 1 *Solution Reproduction Problem*

Phase OFFLINECalcul des solutions haute fidélité $\mathbf{U}^{\text{hf},k}(\mu)$ Construction de la base réduite $\mathbb{Z}_u = [\zeta_{u,j}]$ [eq. (3)]Calcul de la quadrature empirique $\underline{\rho}^{\text{eq}}$ [eq. (5) ou (7)]**Phase ONLINE**Calcul des coefficients de la base réduite des déplacements $\{\Lambda_u^j(t_k, \mu)\}_{j=1}^{N_u}$ Gappy-POD pour le calcul des coefficients de la base réduite des contraintes $\{\Lambda_\sigma^j(t_k, \mu)\}_{j=1}^{N_\sigma}$

$$N_u = \min \left\{ Q \in \mathbb{N}, \quad \sum_{q=1}^Q \lambda_q \geq (1 - \varepsilon_{\text{POD},u})^2 \sum_{q=1}^K \lambda_q \right\}$$

Il est alors possible de définir les modes POD, qui constitueront la base réduite pour les déplacements :

$$\zeta_{u,n} = \sum_{k=1}^K (\psi_n)_k \mathbf{U}^{\text{hf},k}(\mu), \quad \forall n \in \llbracket 1, N \rrbracket$$

Nous notons :

$$\mathbb{Z}_u = \text{POD} \left\{ \left\{ \mathbf{U}^{\text{hf},k}(\mu) \right\}_{k=1}^K, (\cdot, \cdot), \varepsilon_{\text{POD},u} \right\} \quad (3)$$

Nous construisons de la même façon une base réduite associée aux *snapshots* de contraintes à partir d'une tolérance $\varepsilon_{\text{POD},\sigma}$.

3.1.2 Quadrature empirique

Pour un paramètre μ donné, nous cherchons la solution haute-fidélité à partir du résidu sous forme variationnelle $\forall \mathbf{v} \in \mathcal{V}_N, \forall k \in \llbracket 1, K \rrbracket$:

$$\begin{aligned} \mathcal{R}_k(\mathbf{u}^{\text{hf},k}(\mu), \mathbf{v}; \mu) &= \int_{\Omega} \underline{\underline{\sigma}}(\mathbf{u}^{\text{hf},k}(\mu), \mathbf{u}^{\text{hf},k-1}(\mu)) : \underline{\underline{\varepsilon}}(\mathbf{v}) - \mathbf{f} \cdot \mathbf{v} \, dx \\ &:= \int_{\Omega} \eta_k^{\text{lhs}}(\mathbf{u}^{\text{hf},k}(\mu), \mathbf{v}, x; \mu) - \eta_k^{\text{rhs}}(\mathbf{u}^{\text{hf},k}(\mu), \mathbf{v}, x; \mu) \, dx, \end{aligned}$$

Nous introduisons la notation du résidu :

$$R_{i,n,k,m,*} = \eta_k^*(\mathbf{u}^{\text{hf}}(\mu_i), \zeta_{u,n}, x_m^{\text{hf}}; \mu_i)$$

où $i \in \llbracket 1, n_{\text{train}} \rrbracket$ correspond au paramètre i , $n \in \llbracket 1, N_u \rrbracket$ au j -ème vecteur de la base réduite, $k \in \llbracket 1, K \rrbracket$ au temps t_k , $m \in \llbracket 1, \mathcal{N}_G \rrbracket$ correspond au point d'intégration pour la quadrature exacte m , et $*$ $\in \{\text{lhs}, \text{rhs}\}$. Bien plus, en notant $\underline{\rho}^{\text{hf}}$ la quadrature exacte utilisée pour la résolution haute-fidélité :

$$\int_{\Omega} R_{i,n,k,*,*} d\Omega = \sum_{m=1}^{\mathcal{N}_G} \rho_m^{\text{hf}} \eta_k^*(\mathbf{u}(\mu_i), \zeta_{u,n}, x_m^{\text{hf}}; \mu), \quad \text{où } R_{i,n,k,*,*} = \eta_k^*(\mathbf{u}(\mu_i), \zeta_{u,n}, \cdot; \mu)$$

Nous souhaitons disposer d'une loi de quadrature empirique creuse, dont l'ensemble des poids sont positifs ($\underline{\rho} \geq 0$), qui conserve la mesure du domaine et approche les intégrales à une tolérance δ près, c'est-à-dire pour $\in \llbracket 1, n_{\text{train}} \rrbracket, n \in \llbracket 1, N_u \rrbracket, k \in \llbracket 1, K \rrbracket$:

$$\left\{ \begin{array}{l} \left| |\Omega| - \sum_{m=1}^{\mathcal{N}_G} \rho_m \right| \leq \delta |\Omega| \\ \left| \sum_{m=1}^{\mathcal{N}_G} \rho_m R_{i,n,k,*,*} - \int_{\Omega} R_{i,n,k,*,*} \right| \leq \delta \left| \int_{\Omega} R_{i,n,k,*,*} d\Omega \right| \end{array} \right.$$

L'ensemble de ces contraintes permet de reformuler le problème de quadrature empirique comme un problème de minimisation en norme ℓ_0 . Dans la littérature, on parle aussi de problème de représentation creuse.

$$\min_{\underline{\rho} \in \mathcal{N}_{\mathcal{G}}} \|\underline{\rho}\|_{\ell_0} \quad \text{s.t.} \quad \begin{cases} \left| \mathbb{G}\underline{\rho} - \mathbf{y}^{\text{fe}} \right| \leq \delta |\mathbf{y}^{\text{fe}}| \\ \left| \mathbf{1}^T \underline{\rho} - |\Omega| \right| \leq \delta |\Omega| \\ \underline{\rho} \geq 0 \end{cases} \quad (4)$$

où la matrice $\mathbb{G} \in \mathbb{R}^{n_{\text{eq}} \times \mathcal{N}_{\mathcal{G}}}$ est appelée dictionnaire, et son nombre de lignes dépend du nombre d'intégrales (n_{eq}) que nous voulons approcher. L'entier n_{eq} dépendra alors du nombre de pas de temps dans le problème (problème statique ou dynamique), du nombre de vecteurs dans la base réduite, mais aussi du nombre de paramètres explorés ($n_{\text{eq}} = K \times n_{\text{train}} \times N_u$ dans le cas général).

Chaque ligne du dictionnaire correspond à la transposée du vecteur de l'intégrande aux points d'intégration, tandis que le second membre est le vecteur des intégrales. Une ligne du dictionnaire correspond à la transposée du vecteur $[\eta_k^*(\mathbf{u}(\mu), \zeta_{u,n}, x_m^{\text{hf}}; \mu)]_{m=1}^{\mathcal{N}_{\mathcal{G}}}$, tandis que la composante du second membre doit être égal à la valeur exacte de l'intégrale, c'est-à-dire :

$$\int_{\Omega} \eta_k^*(\mathbf{u}(\mu), \zeta_{u,n}, x; \mu) dx = \mathbf{1}^T \cdot [\eta_k(\mathbf{u}(\mu), \zeta_{u,n}, x_m^{\text{hf}}; \mu)]_{m=1}^{\mathcal{N}_{\mathcal{G}}}$$

Le problème (4) est NP-difficile et n'est donc pas résolu directement en pratique. Néanmoins, plusieurs approches alternatives ont été développées pour approcher ce problème. Nous présentons trois algorithmes de résolution, que nous avons testé et validé sur un cas numérique. Une première approche repose sur la relaxation ℓ_1 du problème. Nous parlons aussi de quadrature empirique par programmation linéaire (*LP-EQ* [12]) Le problème de représentation creuse (4) peut donc être relaxé en norme ℓ_1 comme suit :

$$\min_{\underline{\rho} \in \mathcal{N}_{\mathcal{G}}} \|\underline{\rho}\|_{\ell_1} \quad \text{s.t.} \quad \begin{cases} \left| \mathbb{G}\underline{\rho} - \mathbf{y}^{\text{fe}} \right| \leq \delta |\mathbf{y}^{\text{fe}}| \\ \left| \mathbf{1}^T \underline{\rho} - |\Omega| \right| \leq \delta |\Omega| \\ \underline{\rho} \geq 0 \end{cases} \quad (5)$$

Ce dernier peut être alors reformulé sous la forme d'un problème de programmation linéaire :

$$\min_{\underline{\rho} \in \mathcal{N}_{\mathcal{G}}} \mathbf{1}^T \underline{\rho} \quad \text{s.t.} \quad \begin{cases} \mathbb{A}\underline{\rho} \leq \mathbf{b} \\ \underline{\rho} \geq 0 \end{cases} \quad (6)$$

en introduisant les matrices suivantes :

$$\mathbb{A} = \begin{bmatrix} \tilde{\mathbb{G}} & \mathbf{0} \\ \mathbf{0} & -\tilde{\mathbb{G}} \end{bmatrix}, \quad \tilde{\mathbb{G}} = \begin{bmatrix} \mathbb{G} \\ \mathbf{1}^T \end{bmatrix}, \quad \text{et} \quad \mathbf{b} = \begin{bmatrix} \tilde{\mathbf{y}}^{\text{fe}} + \delta \tilde{\mathbf{y}}^{\text{fe}} \\ -\tilde{\mathbf{y}}^{\text{fe}} + \delta \tilde{\mathbf{y}}^{\text{fe}} \end{bmatrix}, \quad \tilde{\mathbf{y}}^{\text{fe}} = \begin{bmatrix} \mathbf{y}^{\text{fe}} \\ |\Omega| \end{bmatrix}$$

Une autre méthode (*ECSW* [7]) consiste à se ramener à un problème de minimisation en norme ℓ_2 . Cette approche se fonde sur une résolution d'un problème aux moindres carrés à poids positifs (*NNLS* : *Non-negative Least-Square*). Cette méthode repose sur la définition d'un espace pour les vecteurs de poids, noté Φ , à partir duquel le problème de représentation creuse peut être formulé comme un problème d'optimisation :

$$\underline{\rho}^{\text{eq}} = \arg \min_{\underline{\rho} \in \Phi} \|\underline{\rho}\|_{\ell_0}, \quad \text{avec} \quad \Phi = \left\{ \underline{\rho} \in \mathbb{R}^{\mathcal{N}_{\mathcal{G}}}, \quad \|\tilde{\mathbb{G}}\underline{\rho} - \tilde{\mathbf{y}}^{\text{fe}}\|_2 \leq \delta \|\tilde{\mathbf{y}}^{\text{fe}}\|_2, \quad \underline{\rho} \geq 0 \right\}$$

La méthode ECSW, qui peut être aussi vu comme une relaxation ℓ_2 , repose sur la substitution de ce problème par un problème NNLS (proposée par [8]) :

$$\underline{\rho}^{\text{EQ}} = \arg \min_{\underline{\rho} \in \Upsilon} \|\tilde{\mathbb{G}}\underline{\rho} - \tilde{\mathbf{y}}^{\text{fe}}\|_2^2, \quad \text{avec} \quad \Upsilon = \left\{ \underline{\rho} \in \mathbb{R}^{6\mathcal{N}_{\mathcal{G}}}, \quad \underline{\rho} \geq 0 \right\} \quad (7)$$

L'algorithme de contraintes actives de Lawson and Haron ([10]) est utilisé et modifié avec un critère d'arrêt supplémentaire, similaire à celui donné pour la relaxation ℓ_1 (équation (6)) :

$$\left\| \tilde{\mathbf{G}}\underline{\mathbf{p}} - \tilde{\mathbf{y}}^{fe} \right\|_2 \leq \delta \left\| \tilde{\mathbf{y}}^{fe} \right\|_2$$

Enfin, une dernière approche algorithmique consiste à résoudre le problème de minimisation en norme ℓ_2 à l'aide d'un algorithme de type *Non-Negative Orthogonal Matching Pursuit* ([3], [4]).

Suite à la résolution du problème (4) par l'une des méthodes présentées, nous disposons d'un vecteur de poids $\underline{\mathbf{p}}^{eq}$ défini sur le maillage total. A partir de ce vecteur, nous pouvons définir un maillage réduit, constitué uniquement des éléments où il existe un poids d'intégration non-nul. Le problème réduit (2) est donc résolu sur ce maillage réduit.

3.1.3 Reconstruction des contraintes par *Gappy-POD*

A la fin d'un appel au calcul réduit, nous disposons des solutions réduites en déplacement $\mathbf{u}_N^k(\mu)$, et des contraintes associées (par la loi de comportement considérée) aux points d'intégration sélectionnés par la quadrature empirique. Néanmoins, ces vecteurs de contraintes n'appartiennent pas, en toute généralité, à l'espace réduit construit pour les contraintes. Nous utilisons alors un algorithme de *Gappy-POD* ([6]) pour déterminer les coefficients réduits $\{\Lambda_\sigma^j(t_k, \mu)\}_{j=1}^{N_\sigma}$.

3.2 Problème paramétrique

L'extension du problème précédent à un problème paramétrique soulève deux enjeux : l'adaptation des méthodes de compression de données et de quadrature empirique à plusieurs paramètres et la construction d'un indicateur d'erreur sur notre modèle. Ce dernier doit être obtenu pour un faible coût de calcul et doit être corrélé à l'erreur entre la solution haute-fidélité et la solution réduite.

3.2.1 Adaptation des étapes précédentes

Pour obtenir une base hiérarchique, nous avons décidé d'utiliser une POD incrémentale [1] [9], en appliquant la procédure de POD sur la projection des nouveaux *snapshots* sur la base existante. Supposons que nous disposons d'une base réduite construite \mathbb{Z}_u et les nouveaux *snapshots* $\{\mathbf{U}^{hf,k}(\mu^*)\}_{k=1}^K$. La nouvelle base est construite par concaténation :

$$\mathbb{Z}_u = [\mathbb{Z}_u, \mathbb{Z}_{proj}], \quad \text{où} \quad \mathbb{Z}_{proj} = \text{POD} \left\{ \left\{ \Pi_{\mathbb{Z}_u^\perp, (\cdot, \cdot)} \mathbf{U}^{hf,k}(\mu) \right\}_{k=1}^K, (\cdot, \cdot), \varepsilon_{\text{POD}, u} \right\}$$

où $\Pi_{\mathbb{Z}_u^\perp, (\cdot, \cdot)}$ est l'opérateur de projection sur \mathbb{Z}_u^\perp au sens du produit scalaire (\cdot, \cdot) . Pour la quadrature empirique, nous utilisons les mêmes algorithmes que ceux présentés dans la partie 3.1.2, où l'ensemble du dictionnaire est reconstruit.

3.2.2 Construction d'un indicateur d'erreur a posteriori

Nous introduisons, dans le cas d'une force extérieure indépendante du temps, un indicateur d'erreur *a posteriori* comme la moyenne temporelle des norme duales des opérateurs définis à chaque pas de temps :

$$\Delta_N = \sqrt{\frac{1}{K} \sum_{k=1}^K (\Delta_N^k)^2}, \quad \text{où} \quad \Delta_N^k(\mu) = \sup_{\mathbf{v} \in \mathcal{V}^{hf}, \|\mathbf{v}\|_{\mathcal{V}^{hf}}=1} \left[\sum_{n=1}^{N_\sigma} \Lambda_\sigma^n(t_k, \mu) \int_\Omega \zeta_{\sigma,n} : \varepsilon(\mathbf{v}) \, d\Omega - \int_\Omega \mathbf{f} \cdot \mathbf{v} \, d\Omega, \right]$$

L'estimation de cette quantité peut se reformuler à l'aide des représentants de Riesz des formes linéaires qui apparaissent dans la formulation. Nous pouvons calculer ces représentants $\{\Psi_n^\sigma\}_{n=1}^{N_\sigma+1}$ définis comme suit $\forall \mathbf{v} \in \mathcal{V}^{hf}, \forall n \in \llbracket 1, N_\sigma + 1 \rrbracket$:

$$(\Psi_n^\sigma, \mathbf{v})_{\mathcal{V}} = \mathcal{L}_n(\mathbf{v}) \quad \text{où} \quad \begin{cases} \mathcal{L}_n(\mathbf{v}) = \int_\Omega \zeta_{\sigma,n} : \varepsilon(\mathbf{v}) \, d\Omega, & \forall n \in \llbracket 1, N_\sigma \rrbracket \\ \mathcal{L}_{N_\sigma+1}(\mathbf{v}) = \int_\Omega \mathbf{f} \cdot \mathbf{v} \, d\Omega \end{cases}$$

Il est alors possible de définir un indicateur d'erreur, écrit sous la forme compacte :

$$\Delta_N(\mu) = \sqrt{\frac{1}{K} \sum_{k=1}^K [\Lambda_{\sigma,k}(\mu)]^T \cdot \Sigma_N \cdot \Lambda_{\sigma,k}(\mu)}, \quad \text{où } \Lambda_{\sigma,k}(\mu) = \begin{pmatrix} \Lambda_{\sigma}^1(t_k, \mu) \\ \dots \\ \Lambda_{\sigma}^{N_{\text{POD},\sigma}}(t_k, \mu) \\ -1 \end{pmatrix}$$

et $(\Sigma_N)_{n,m} = (\Psi_n^\sigma, \Psi_m^\sigma)_{\mathcal{V}}$ est une matrice qui peut être pré-calculée durant la phase *offline*. L'indicateur introduit nécessite durant la phase *online* de disposer des vecteurs de coordonnées réduites et d'effectuer deux multiplications matricielles par pas de temps. Le prochain paramètre exploré est celui pour lequel l'estimateur est le plus élevé :

$$\Delta_N^{\max} = \max_{\mu \in \Theta} \Delta_N(\mu), \quad \text{et } \mu^* = \arg \max_{\mu \in \Theta} \Delta_N(\mu)$$

4 Cas de validation et résultats numériques

4.1 Remarques sur l'implémentation générale

Notre démarche a été d'implémenter notre approche à l'aide d'une librairie Python appelé Mordicus, où nous avons utilisé, modifié (calculs de modes POD) ou développé les fonctionnalités spécifiques à la réduction de modèles (quadrature empirique, calcul d'indicateur d'erreur). Ce code peut être interfacé de manière non-intrusive à un code éléments finis (Code Aster [5]), de manière à construire un modèle réduit adéquat. Le code développé n'a pas de connaissance a priori des lois de comportements mécaniques, et extrait les vecteurs de déplacements et de contraintes à partir des fichiers de résultats de Code Aster. L'ensemble de l'algorithme a été mis en place et construit pour considérer des lois mécaniques non-linéaires, mais la validation présentée ici se limite au cas de l'élasticité linéaire isotrope.

4.2 Cas de validation

Notre implémentation dans Code Aster a été testée sur un cas d'élasticité linéaire en considérant le coefficient de Poisson ν comme paramètre variable :

$$\underline{\underline{\sigma}} = \frac{E}{1+\nu} \left(\underline{\underline{\varepsilon}} + \frac{\nu}{1-2\nu} \text{Tr}(\underline{\underline{\varepsilon}}) \underline{\underline{I}} \right)$$

Nous considérons une poutre 3D encastée soumise à une force volumique. Le problème est discrétisé avec 8804 éléments hexaédriques, avec 20 noeuds par élément, et 27 points d'intégration (237708 points d'intégration au total). Nous avons testé nos algorithmes sur un intervalle variable $\mu = \nu \in [0.1; 0.4]$, où nous considérons un ensemble Θ de six paramètres :

$$\Theta = \{0.1, 0.16, 0.22, 0.28, 0.34, 0.44\}$$

Le produit scalaire utilisé dans la méthode POD et les erreurs en normes relatives ont été calculées en utilisant le produit scalaire ℓ_2 sur les vecteurs sortis par le logiciel Code Aster (déplacements et contraintes). Les erreurs relatives sur les déplacements (figure 1) et les contraintes sont consistants avec les ordres de grandeur de la tolérance δ .

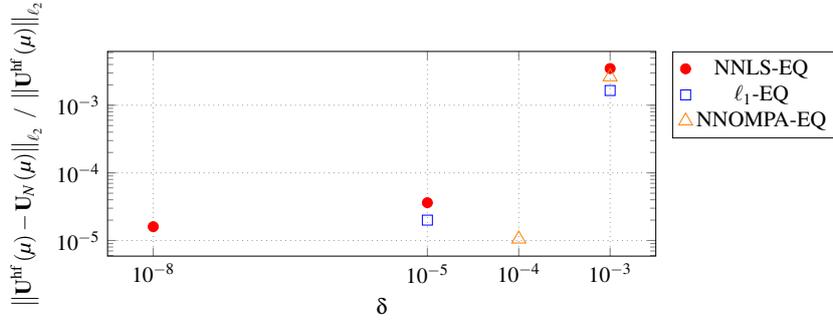


FIGURE 1 – Erreur relative sur les déplacements (norme ℓ_2)

Bien plus, la procédure de quadrature empirique permet de réduire drastiquement la taille du maillage. Sur les 237708 points d'intégration, toutes les méthodes utilisées nous ramènent à utiliser quelques dizaines de points d'intégration (80 au maximum durant nos simulations). Le rapport des temps de résolution du problème mécanique entre le solveur haute-fidélité et le solveur réduit avec quadrature empirique illustre la diminution du temps de calcul (plus de 300 fois plus long) obtenue grâce à cette approche (voir tableau 1).

	Solveur HF	Solveur Réduit (sans EQ)	Solveur réduit (ℓ_1 -EQ)
Temps de calcul CPU (s)	16.9	4.6	0.045
Pourcentage du temps CPU HF	–	27%	0.27%

TABLE 1 – Speedup du solveur réduit

Quelle que soit la méthode de quadrature empirique adoptée, les algorithmes utilisés permettent d'observer une décroissance de notre indicateur d'erreur (figures 2 et 3). Ce dernier atteint au bout de plusieurs itérations un palier correspond au palier fixé par notre tolérance δ . La convergence de l'indicateur vers le zéro numérique valide la construction et l'implémentation de cette quantité.

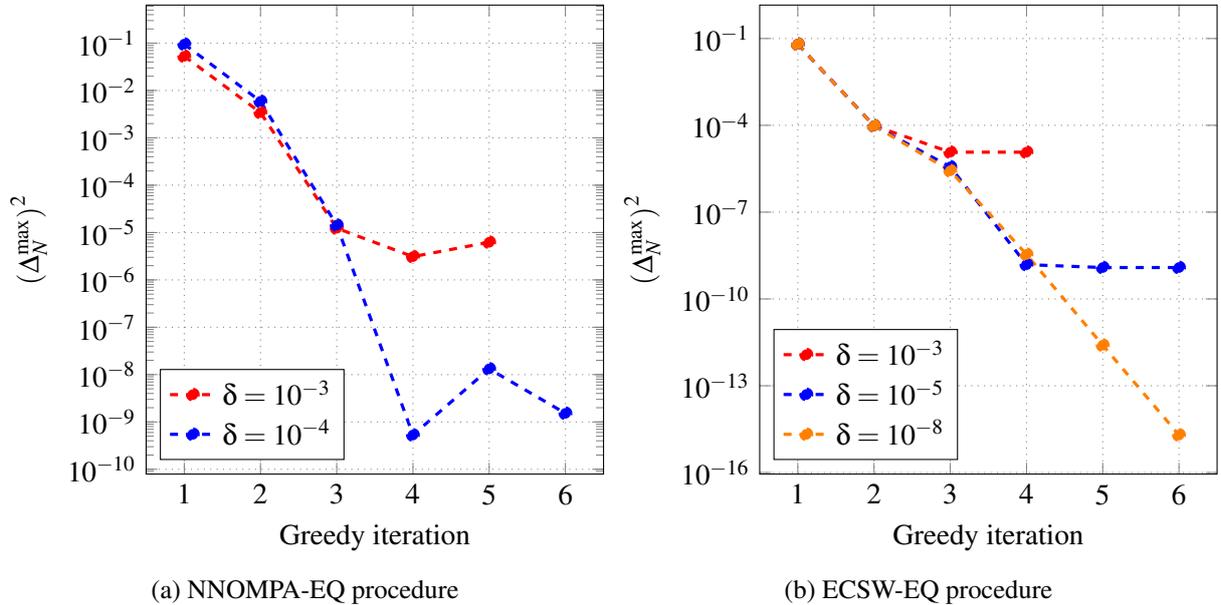


FIGURE 2 – Indicateur pour des méthodes de quadrature empirique de relaxation ℓ_2

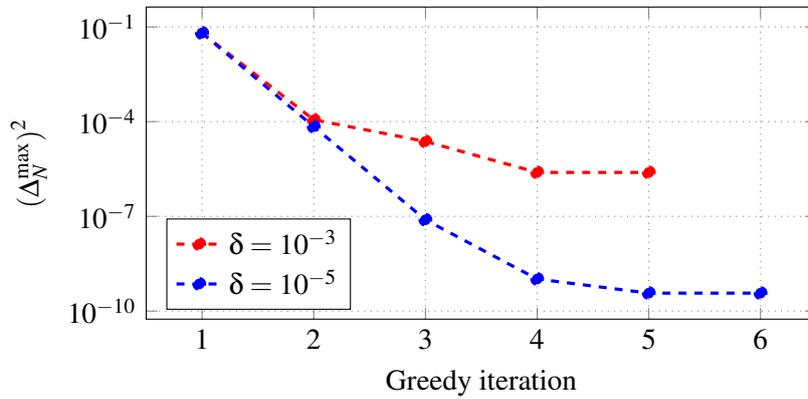


FIGURE 3 – Indicateur pour la méthode de quadrature empirique de relaxation ℓ_1

5 Conclusion et perspectives

Les premiers résultats obtenus semblent démontrer l'efficacité de la construction de l'algorithme glouton (*POD-Greedy*). Les erreurs relatives obtenues sur les variables d'intérêt (déplacements, contraintes) et la réduction du temps de calcul vont dans ce sens. Ce travail va être prolongé sur des calculs non-linéaires pour tester la robustesse des algorithmes présentés. Plusieurs développements sont en cours pour permettre d'étendre ce cas à un cas industriel (calculs sur des maillages de taille plus importante, prise en compte d'une quadrature empirique surfacique, prise en compte de conditions aux limites sous la forme de multiplicateurs de Lagrange).

Références

- [1] A. Benaceur, *Réduction de modèles en thermo-mécanique*. Thèse de doctorat en Mathématiques. Paris-Est. 2018
- [2] P. Benner, S. Gugercin, K. Willcox. *A survey of projection based methods for parametric dynamical systems*. Society for Industrial and Applied Mathematics. SIAM review 57, no 4 : 483-531. 2015
- [3] F. Casenave, N. Akkari, F. Bordeu, C. Rey, D. Ryckelynck. *A nonintrusive distributed reduced-order modeling framework for nonlinear structural mechanics—Application to elastoviscoplastic computations*. International Journal for Numerical Methods in Engineering, 121(1), 32-53, 2020
- [4] F. Casenave, N. Akkari. *An error indicator-based adaptive reduced order model for nonlinear structural mechanics—application to high-pressure turbine blades*. Mathematical and computational applications, 24(2), 41, 2019.
- [5] EDF R&D. *Code Aster*, logiciel sous licence GNU GPL, www.code-aster.org.
- [6] R. Everson. and L. Sirovich. *Karhunen–loève procedure for gappy data*. JOSA A, 12(8) :1657–1664, 1995
- [7] C. Farhat, T. Chapman, P. Avery. *Structure-preserving, stability, and accuracy properties of the energy-conserving sampling and weighting method for the hyper reduction of nonlinear finite element dynamic models*, International journal for numerical methods in engineering, Wiley Online Library, 102(5) :1077–1110, 2015
- [8] C. Farhat, P. Avery, T. Chapman, J. Cortial. *Dimensional reduction of nonlinear finite element dynamic models with finite rotations and energy-based mesh sampling and weighting for computational efficiency*, International Journal for Numerical Methods in Engineering, , Wiley Online Library, 98(9) : 625–662, 2014
- [9] L. Khoun. *Réduction de modèles pour les problèmes vibro-acoustique transitoires paramétriques—Application aux problèmes de pré-dimensionnement de structures immergées aux ondes de choc d'explosion*. Thèse de doctorat en Mathématiques. 2021
- [10] C. Lawson, R.J. Hanson. *Solving Least Squares Problems*, SIAM, 1987
- [11] L. Sirovich. *Turbulence and the dynamics of coherent structures. I. Coherent structures*. Quarterly of applied mathematics, 45(3), 561-571, 1987
- [12] M. Yano, A. Patera, *An LP empirical quadrature procedure for reduced basis treatment of parametrized nonlinear PDEs*, Computer Methods in Applied Mechanics and Engineering, Elsevier, 1104–1123, 2019.