

Résolution de problèmes non linéaires et instationnaires dans un code industriel à l'aide d'une méthode de réduction de modèles faiblement intrusive

P.-E.Malleval^{1,2}, D. Néron², R. Scanff¹

¹ Siemens Industry Software SAS, 92320 Châtillon, France.
{pierre-eliot.malleval, ronan.scanff}@siemens.com

² Université Paris-Saclay, CentraleSupélec, ENS Paris-Saclay, CNRS, LMPS - Laboratoire de mécanique Paris-Saclay, 91190, Gif-sur-Yvette, France.
{pierre-eliot.malleval, david.neron}@ens-paris-saclay.fr

Résumé — L'intégration de techniques de réduction de modèles dans des logiciels industriels permet de tirer pleinement parti de la gestion de la complexité qu'offrent ces environnements de travail en fournissant simultanément un cadre général, valable pour tous types d'applications. Dans ce travail, une version faiblement intrusive de la méthode LATIN-PGD est étendue au cas des problèmes transitoires dans le logiciel éléments finis généraliste Simcenter Samcef développé par Siemens Digital Industries Software. L'approche est illustrée par des cas industriels portant sur la thermique non linéaire instationnaire.

Mots clés — : réduction de modèles, non linéaire, transitoire, LATIN-PGD, faiblement intrusif

1 Introduction

La simulation numérique occupe aujourd'hui une place essentielle dans de nombreuses branches de l'ingénierie. L'augmentation constante de la puissance et des moyens de calcul peinent cependant à compenser la complexité croissante des modèles que les ingénieurs souhaiteraient pouvoir manipuler. En particulier, on ne peut que constater une explosion de la complexité calculatoire nécessaire aux simulations pour répondre aux enjeux des nouveaux matériaux et moyens de production tels que les composites ou la fabrication additive. Les calculs thermiques et mécaniques sous-jacents à la simulation numérique de ces structures nécessitent de résoudre des problèmes fortement non linéaires, à plusieurs millions de degrés de liberté, intégrant différentes échelles, faisant intervenir différentes physiques et nécessitant une discrétisation temporelle souvent fine. A cela s'ajoute fréquemment la volonté d'explorer un espace paramétrique afin de répondre aux problèmes de conception ou d'optimisation. La nécessité d'obtenir des résultats dans des délais courts, compatibles avec les besoins industriels, conduit alors à une préoccupation croissante pour la réduction du temps de calcul.

Dans ce contexte, les modèles réduits peuvent apporter une valeur substantielle, tant pour la résolution d'un problème complexe que pour la réalisation d'un grand nombre de calculs pour des familles de chargements ou des plages de valeurs de paramètres, en réduisant la complexité apparente du problème à résoudre tout en préservant la qualité du modèle [1]. Deux grandes familles d'approches se distinguent. La première famille est basée sur la construction préalable (ou apprentissage) d'une base réduite émanant d'un grand nombre de résolutions hautes fidélités choisies judicieusement : méthodes Proper Orthogonal Decomposition (POD) [3] ou Reduced Basis (RB) [11, 12]. La seconde famille consiste au contraire à générer la base réduite à la volée à l'aide d'algorithmes adaptés qui permettent de trouver la solution simultanément à la recherche de la base réduite pour représenter celle-ci : c'est le cas de la méthode Proper Generalized Decomposition (PGD) [5, 6].

Malgré un niveau de maturité important et un large éventail d'applications, le développement de ces méthodes dans le secteur industriel est freiné par leur difficulté d'intégration au sein des codes éléments finis commerciaux. En effet, ces méthodes sont souvent considérées comme fortement intrusives ; les algorithmes associés nécessitant des opérations et formats de stockage de données atypiques pour les codes éléments finis industriels. Toutefois, l'attention s'est portée depuis quelques années, sur le développement de méthodes de réductions de modèles moins intrusives. La mise en place des approches a

posteriori POD ou RB dans un cadre industriel ont constitués un premier effort vers l'introduction de ces méthodes dans le quotidien des ingénieurs, comme par exemple dans [2]. En raison de la formulation *a priori* propre à la PGD et plus intrusive, les premières approches basées sur cette méthode restent calquées sur l'utilisation de *snapshots*, par exemple [8], ou reposent sur un couplage externe entre un logiciel industriel et un code développé en laboratoire, comme dans [14]. Ces applications demeurent spécifiques et difficilement intégrables dans un logiciel éléments finis généraliste. Plus récemment, une version faiblement intrusive de la méthode LATIN-PGD, définie au moyen de quantités généralisées, a été élaborée et intégrée nativement dans le logiciel industriel Simcenter Samcef de Siemens [13].

La mise en place de techniques de réduction de modèles au plus proche du code source permet la construction d'un modèle réduit en exploitant pleinement les procédures optimisées du solveur industriel et en accédant à tous les outils déjà présents dans celui-ci : tous types de non-linéarités, d'éléments, de chargements, etc. De plus, l'approche faiblement intrusive s'intègre de manière transparente dans le quotidien des ingénieurs, sans modifier le *workflow* des bureaux d'étude. L'utilisateur bénéficie toujours de toutes les fonctionnalités intégrées dans le logiciel industriel, telles que l'accès aux outils de conception et fabrication assistées par ordinateur, de gestion des assemblages, de maillage, d'optimisation paramétrique, ou encore de continuité numérique, qui facilitent la construction, la mise en donnée et l'analyse de modèles complexes offrant ainsi une solution globale pour résoudre des problèmes d'ingénierie. Cette approche a conduit à une réduction significative des temps de calculs associés aux problèmes mécaniques non linéaires quasi-statiques usuellement rencontrés en environnement industriel. Le travail présenté propose l'extension de cette méthodologie faiblement intrusive pour le traitement de problèmes transitoires, ce qui pose de nouvelles difficultés en terme d'intégration au sein du code en minimisant les modifications à apporter. En particulier, la résolution des problèmes thermiques non linéaires instationnaires susceptibles d'être traités par le logiciel industriel a été étudiée.

2 Formulation du problème thermique non linéaire

2.1 Problème de référence

Nous considérons une structure définie par son domaine $\Omega \subset \mathbb{R}^3$, dont la température évolue sous l'action de diverses sollicitations au cours d'un intervalle de temps $I \subset \mathbb{R}_+$. Différentes sources de non-linéarités dépendantes du temps peuvent intervenir : propriétés matériaux (masse volumique, conductivité thermique, capacité thermique), conditions limites de convection ou de rayonnement, changements de phase, etc. Ce type de problème est communément résolu par la majorité des logiciels éléments finis généralistes commerciaux. La construction du problème éléments finis associé peut être reformulée comme suit, où l'on note $\underline{\mathbf{T}} \in I_{\mathcal{V}} \equiv L^2(I; \mathcal{V})$ les températures généralisées sur l'espace éléments finis $\mathcal{V} \equiv \mathbb{R}^n$ avec n le nombre de degrés de liberté :

Trouver $\underline{\mathbf{T}} \in I_{\mathcal{V}}$ vérifiant :

- Les conditions aux limites en température $\underline{\mathbf{T}}_d(t)$ et la condition initiale $\underline{\mathbf{T}}_0(t)$:

$$\forall t \in I, \quad C(\underline{\mathbf{T}}(t)) = \underline{\mathbf{T}}_d(t) \text{ et } \underline{\mathbf{T}}|_{t=0} = \underline{\mathbf{T}}_0(t) \quad (1)$$

- L'équation d'équilibre thermique entre les contributions des différents flux thermiques :

$$\forall t \in I, \quad \underline{\underline{Q}}^{\text{int}}(t, \underline{\mathbf{T}}(t)) + \underline{\underline{Q}}^{\text{ine}}(t, \dot{\underline{\mathbf{T}}}(t)) - \underline{\underline{Q}}^{\text{ext}}(t) = \underline{\underline{0}} \quad (2)$$

- Les relations non linéaires de transferts thermiques dont résultent les flux généralisées :

$$\forall t \in I, \quad \begin{cases} \underline{\underline{Q}}^{\text{int}}(t, \underline{\mathbf{T}}(t)) = \mathcal{A}_T(t, \underline{\mathbf{T}}(t_{\text{prec}} \leq t)) \\ \underline{\underline{Q}}^{\text{ine}}(t, \dot{\underline{\mathbf{T}}}(t)) = \mathcal{A}_h(t, \dot{\underline{\mathbf{T}}}(t_{\text{prec}} \leq t)) \end{cases} \quad (3)$$

Les températures généralisées $\underline{\mathbf{T}}$ sont l'analogue des déplacements généralisés en mécanique. Nous introduisons le flux *inertiel* de chaleur $\underline{\underline{Q}}^{\text{ine}} \in I_{\mathcal{V}}$, associé au caractère instationnaire du problème traité, c'est à dire dans le cas de l'équation de la chaleur au taux de variation temporelle de l'enthalpie du système. L'opérateur \mathcal{A}_h formalise l'équation d'évolution liant la variation temporelle de température au stockage de chaleur par le matériaux, faisant intervenir la masse volumique et la capacité thermique massique. Le flux *interne* de chaleur $\underline{\underline{Q}}^{\text{int}} \in I_{\mathcal{V}}$, correspond aux transferts thermiques, que ce soit par

conduction, convection ou rayonnement. Ainsi, l'opérateur \mathcal{A}_T formalise la relation reliant le champ de température aux flux de chaleur transmis. Les flux généralisés sont équivalents à des efforts généralisés dans le cadre de la mécanique du solide. L'opérateur \mathcal{C} est linéaire tandis que les opérateurs \mathcal{A}_T et \mathcal{A}_h sont a priori non linéaires. En particulier, ces opérateurs non linéaires se rapportent non seulement au temps, mais aussi à l'historique des températures dans la mesure où la diffusion est un processus majoritairement irréversible. L'état au temps $t \in \mathbb{R}^+$ nécessite le calcul de l'historique pour tous les pas de temps précédents $t_{prec} \leq t$. L'opérateur \mathcal{C} et les opérateurs \mathcal{A}_T et \mathcal{A}_h sont construits à partir du traitement des éléments locaux par le logiciel industriel considéré. Ils agiront comme des boîtes noires dans notre approche, comme expliqué dans [13].

2.2 Méthodes de résolution non linéaire

Classiquement, la résolution du système (1, 2, 3) d'équations non linéaires consiste à résoudre une succession de systèmes linéaires au sein d'un schéma itératif convergent. Pour la très grande majorité des logiciels industriels, un algorithme de Newton-Raphson est utilisé. En pratique, l'intervalle de temps I est découpé en petits incréments sur chacun desquels un problème non linéaire stationnaire est résolu. Le champ de température $\mathbb{T}(t)$ est recherché de manière itérative à chaque pas de temps en annulant le développement au premier ordre du résidu d'équilibre autour de l'itéré précédent. La linéarisation de ces équations amène finalement à la résolution d'une série de problèmes linéaires, jusqu'à convergence. Connaissant la solution au pas de temps t_{p-1} , la solution au pas de temps suivant t_p est recherchée en annulant le résidu d'équilibre \mathcal{R} , défini à partir de l'équation (2). L'annulation de l'approximation à l'ordre un du résidu d'équilibre autour de l'itéré précédent $\mathbb{T}_p^{(k)}$ permet le calcul de l'itéré suivant $\mathbb{T}_p^{(k+1)}$. La résolution consiste alors à trouver la correction $\Delta\mathbb{T}_p^{(k+1)}$ tel que :

$$\begin{cases} D\mathcal{R}(\mathbb{T}_p^{(k)}, \mathbb{T}_{p-1}) \cdot \Delta\mathbb{T}_p^{(k+1)} = -\mathcal{R}(\mathbb{T}_p^{(k)}, \mathbb{T}_{p-1}) \\ \mathbb{T}_p^{(k+1)} = \mathbb{T}_p^{(k)} + \Delta\mathbb{T}_p^{(k+1)} \end{cases} \quad (4)$$

où $D\mathcal{R}$ est la dérivée de \mathcal{R} par rapport à $\mathbb{T}_p^{(k)}$.

Dans le cadre de la résolution des problèmes non linéaires à l'aide de méthodes de réduction de modèles, d'autres approches peuvent être exploitées pour linéariser les équations à résoudre, par exemple les méthodes asymptotiques numériques (ANM) en combinaison avec la POD ou la PGD [4]. Les travaux présentés se placent dans un autre paradigme en utilisant la méthode de résolution non linéaire LATIN [10] définie par une approche non-incrémentale et favorable à la mise en œuvre de la technique PGD.

3 Mise en place du solveur non linéaire LATIN-PGD

La version de la méthode LATIN-PGD présentée dans ce paragraphe, ne s'inscrit pas dans la continuité des premières approches à variables internes mais se présente comme une extension faiblement intrusive des variantes fonctionnelles [13].

3.1 Structure générale de l'algorithme

Les équations non linéaires (1, 2, 3) sont séparées en deux groupes : la variété A_d regroupe les équations (1) et (2) linéaires, tandis que la variété Γ rassemble les équations locales, non linéaires, associées à (3). Un schéma itératif à deux directions de recherche permet la recherche alternative d'une solution vérifiant les équations d'équilibre, appartenant à la variété A_d : étape globale ou linéaire ; puis d'une solution vérifiant le comportement local, non linéaire, appartenant à la variété Γ : étape locale. L'alternance entre les deux étapes est répétée jusqu'à atteindre une convergence à l'intersection des deux espaces $s_\chi \in A_d \cap \Gamma$, voir Figure 1.

$$s_0 \in A_d \dashrightarrow \underbrace{s_\ell \in A_d \xrightarrow{\text{locale}} \hat{s}_\ell \in \Gamma \xrightarrow{\text{linéaire}} s_{\ell+1} \in A_d}_{\text{Itération } \ell} \dashrightarrow s_\chi \in A_d \cap \Gamma$$

FIGURE 1 – Schéma itératif convergent méthode LATIN.

Une des particularités de l’approche utilisée est le choix d’une direction de recherche verticale au niveau de l’étape locale (A_d vers Γ), traduite par $(\underline{T}_\ell, \underline{\dot{T}}_\ell) = (\hat{\underline{T}}_\ell, \hat{\underline{\dot{T}}}_\ell)$. Ce choix, ainsi que la séparation des difficultés associée à la constitution des variétés A_d et Γ , autorisent une résolution très proche de l’algorithme de Newton-Raphson structurant le code industriel. Au lieu de résoudre à chaque pas de temps, de manière itérative, un problème linéarisé issu de la restriction des variétés A_d et Γ au pas de temps t_p (algorithme de Newton-Raphson), les boucles sur les itérations et sur les pas de temps sont permuées, conduisant à la résolution d’un problème linéarisé global en temps à chaque itération (algorithme LATIN).

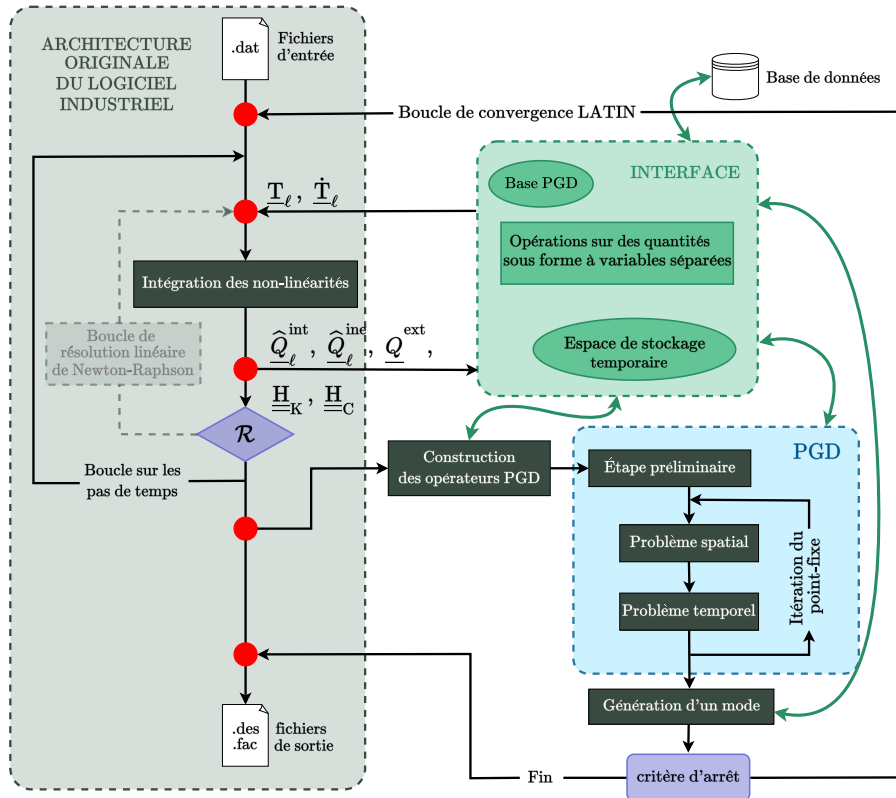


FIGURE 2 – Architecture de la méthodologie faiblement intrusive mise en place dans le code source du logiciel éléments finis industriel Simcenter Samcef - adaptée de [13].

3.2 Étape locale

L’étape locale est traitée en *boîte noire*, comme illustré sur la Figure 2 : les champs $\underline{T}_\ell(t)$ et $\underline{\dot{T}}_\ell(t)$ sont fournis en entrée et la sortie renvoie les flux de chaleur généralisés $\hat{Q}_\ell^{int}, \hat{Q}_\ell^{ine}, Q_\ell^{ext}$ ainsi que les opérateurs nécessaires à la construction de la direction de recherche. L’intégration du comportement et des non-linéarités dissimulés derrière les opérateurs \mathcal{A}_T et \mathcal{A}_h , voir équation (3), s’effectue en chaque point de Gauss de chaque éléments et à chaque pas de temps. C’est une étape classiquement effectuée dans les codes commerciaux à un pas de temps fixé. Une modification des critères d’arrêts permet de réaliser cette intégration sur l’ensemble de l’intervalle de temps sans mettre œuvre la boucle de résolution linéaire interne associée à l’algorithme de Newton-Raphson. Ainsi, on entreprend systématiquement, après intégration du comportement, la gestion du pas de temps suivant. Dès lors, il est possible de réuti-

liser les routines et algorithmes internes développés dans les codes éléments finis et ainsi d’englober toute la richesse du logiciel industriel (tous types d’éléments, toutes lois de comportement, etc.) pour la construction des opérateurs \mathcal{A}_T et \mathcal{A}_h .

3.3 Étape globale

La seconde étape, dite globale ou linéaire, consiste à rechercher la solution $s_{\ell+1} \in A_d$ connaissant $\widehat{s}_\ell \in \Gamma$ en utilisant la direction de recherche en descente \mathbb{H} . La méthode LATIN conduit, lors de l’étape globale, à l’écriture d’un problème constitué d’un ensemble d’équations linéaires sur l’intervalle espace-temps complet. Le fait de travailler sur l’ensemble de l’espace et du temps permet d’utiliser une représentation intelligente sous forme à variables séparées en mettant en œuvre la PGD. L’approximation PGD consiste à écrire la solution \underline{T}_ℓ sous la forme d’une somme finie de fonctions à variables séparées du type (5) :

$$\underline{T}_{\ell+1}(t) \approx \underline{T}_{\ell+1}^{m_{\ell+1}}(t) = \sum_{i=1}^{m_{\ell+1}} \lambda_i(t) \underline{\Delta}_i + \underline{T}_0(t) \quad (5)$$

où les conditions aux limites non homogènes sont prises en compte au moyen d’un relèvement à partir de l’approximation \underline{T}_0 issue de l’initialisation s_0 . Cette étape linéaire n’est pas classique dans les codes industriels puisqu’elle nécessite de manipuler des quantités non convergées sur l’ensemble de l’espace et du temps. Elle requiert ainsi la création d’une « interface » interne, visible sur la Figure 2, pour la gestion de ces quantités. Les routines propres à la recherche d’une correction sous forme PGD sont également ajoutées au code source du logiciel industriel.

L’extension de la méthode pour les problèmes instationnaires entraîne des modifications par rapport au cas quasi-statiques traités précédemment [13]. Dans cette variante, un couple supplémentaire de quantités d’intérêts : $(\underline{T}_\ell, \underline{Q}_\ell^{\text{int}})$ est considéré. Une deuxième direction de recherche est alors nécessaire pour traiter spécifiquement ces variables, conduisant lors de l’étape globale à la recherche de $s_{\ell+1} \in A_d$, vérifiant les équations (1) et (2), tel que :

$$\forall t \in I, \quad \begin{cases} \underline{H}_K(t) \Delta \underline{T}_{\ell+1}(t) = \underline{Q}_{\ell+1}^{\text{int}}(t) - \widehat{Q}_\ell^{\text{int}}(t) \\ \underline{H}_C(t) \Delta \dot{\underline{T}}_{\ell+1}(t) = \underline{Q}_{\ell+1}^{\text{ine}}(t) - \widehat{Q}_\ell^{\text{ine}}(t) \\ \underline{T}_{\ell+1}(t) = \widehat{\underline{T}}_\ell(t) + \Delta \underline{T}_{\ell+1}(t) \end{cases}$$

Le calcul des opérateurs \underline{H}_K et \underline{H}_C associés à la direction de recherche \mathbb{H} est mené au niveau de l’étape locale précédente, en même temps que les flux internes et inertiels généralisés. La correction recherchée vérifie alors l’équation suivante :

$$\underline{H}_C(t) \Delta \dot{\underline{T}}_{\ell+1}(t) + \underline{H}_K(t) \Delta \underline{T}_{\ell+1}(t) = \underline{Q}^{\text{ext}}(t) - \widehat{Q}_\ell^{\text{int}}(t) - \widehat{Q}_\ell^{\text{ine}}(t)$$

L’apparition du terme instationnaire $\underline{H}_C \Delta \dot{\underline{T}}_{\ell+1}$ modifie la nature des équations résolues par la méthode PGD et impose la modification des routines associées ; notamment les étapes de mise à jour et de génération des modes.

4 Illustration sur un cas test

4.1 Description du cas test

Le cas test, décrit sur la Figure 3, concerne une ailette refroidie par air et à l’aide d’un fluide caloporteur circulant dans un perçage cylindrique. Les températures de l’air ambiant et du fluide de refroidissement, respectivement T_e et T_f , sont considérées constantes. Une température T_d est imposée au pied de l’ailette et les bords de l’ailette sont soumis à des conditions limites non linéaires de convection et de radiation. Les surfaces non marquées sur la Figure 3 sont isolées thermiquement. Le débit du fluide caloporteur varie au cours du temps, cette variation est retranscrite par une évolution temporelle du coefficient de convection entre le fluide et la paroi du perçage, voir Figure 4b. Les propriétés thermiques du métal dépendent de la température, comme illustré sur la Figure 4a. La réalisation du cas test est facilitée par l’utilisation de l’environnement industriel Simcenter 3D, qui offre une interface unique permettant d’effectuer toutes les étapes, du pré-traitement au post-traitement. La résolution du problème thermique est

effectuée avec le solveur Simcenter Samcef LATIN-PGD nativement présent dans le logiciel industriel. La structure possède 70 000 degrés de liberté et la discrétisation temporelle compte 240 pas de temps.

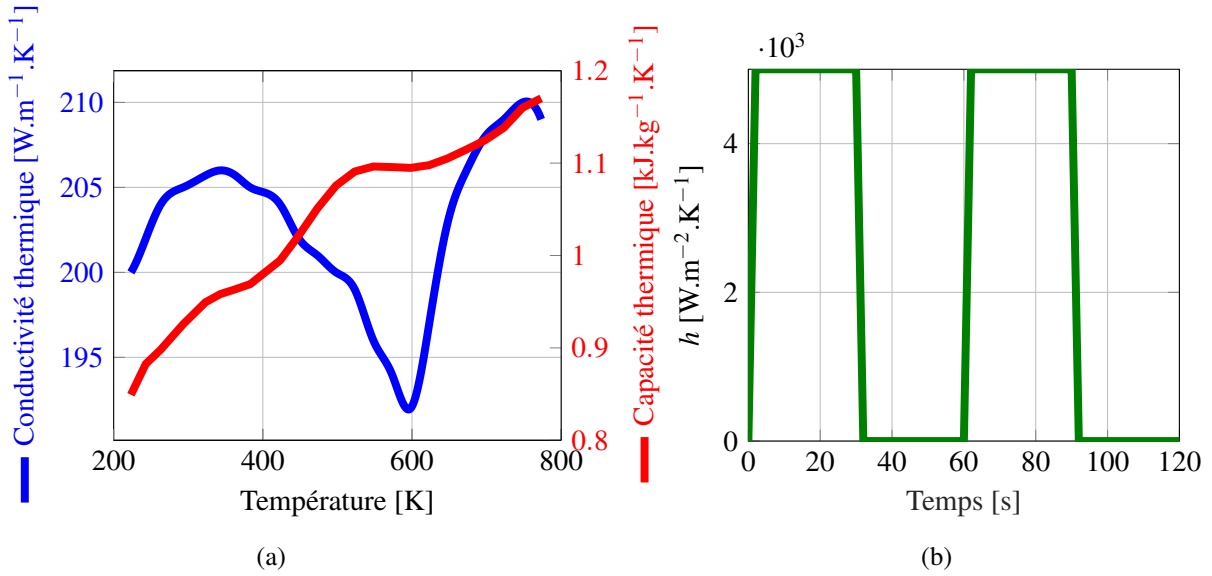


FIGURE 4 – Propriétés matériaux (a) et évolution temporelle du coefficient de convection (b).

4.2 Résultats numériques

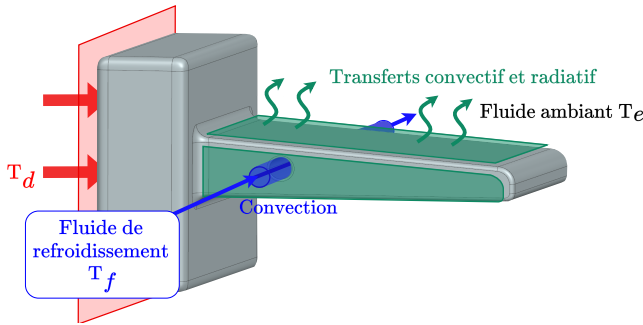


FIGURE 3 – Cas test : ailette de refroidissement 3D.

Ce paragraphe se concentre principalement sur l'évaluation de la méthode LATIN-PGD pour le problème considéré, en décrivant son comportement global et sa vitesse de convergence. Le cas test, décrit dans le paragraphe précédent, est conçu comme une preuve de concept de la méthodologie faiblement intrusive étendue aux problèmes de thermique instationnaire. Dans cette démarche, l'objectif est de réaliser l'ensemble du calcul à l'intérieur de Simcenter 3D, sans recourir à des outils extérieurs. Cette approche a pour but de démontrer la viabilité

et l'efficacité de la méthodologie dans un contexte industriel, renforçant ainsi la confiance dans sa robustesse et sa capacité à être mise en œuvre de manière autonome.

Les résultats obtenus avec la méthode LATIN-PGD sont comparés à une solution de référence obtenue dans Simcenter Samcef avec la méthode de Newton-Raphson. Plus précisément, la Figure 5 ci-dessous, représente la décroissance de l'erreur ε_χ en fonction du nombre d'itérations. Le calcul de l'indicateur d'erreur ω_ℓ permet d'estimer l'état de convergence du calcul. L'indicateur d'erreur est basé sur une norme énergétique moyennée sur l'intervalle de temps. Ce critère permet de mesurer la distance géométrique entre les espaces Γ et A_d et pilote l'arrêt du calcul. L'erreur par rapport à la référence ε_χ est définie comme la différence des champs de température au sens de la norme suivante :

$$\left\{ \begin{array}{l} \omega_\ell = \left[\frac{\|\underline{T}_{\ell+1} - \hat{\underline{T}}_\ell\|_{\mathbb{N}}^2}{\frac{1}{2}\|\underline{T}_{\ell+1}\|_{\mathbb{N}}^2 + \frac{1}{2}\|\hat{\underline{T}}_\ell\|_{\mathbb{N}}^2} \right]^{0.5} \\ \varepsilon_\chi = \frac{\|\underline{T}_\ell - \underline{T}_\chi\|_{\mathbb{N}}}{\|\underline{T}_\chi\|_{\mathbb{N}}} \end{array} \right. \quad \text{avec} \quad \|\underline{T}_\ell\|_{\mathbb{N}}^2 = \langle \underline{T}_\ell, \mathbb{N}\underline{T}_\ell \rangle_{I_v} = \int_I {}^t \underline{T}_\ell(t) \mathbb{N} \underline{T}_\ell(t) dt \quad (6)$$

où \underline{T}_χ représente la solution de référence et l'opérateur \mathbb{N} résulte de la discrétisation et de la combinaison des directions de recherche \underline{H}_K et \underline{H}_C . On observe qu'une valeur de l'indicateur d'erreur de 0.002 coïncide avec une erreur vraie de l'ordre de 0.1% :

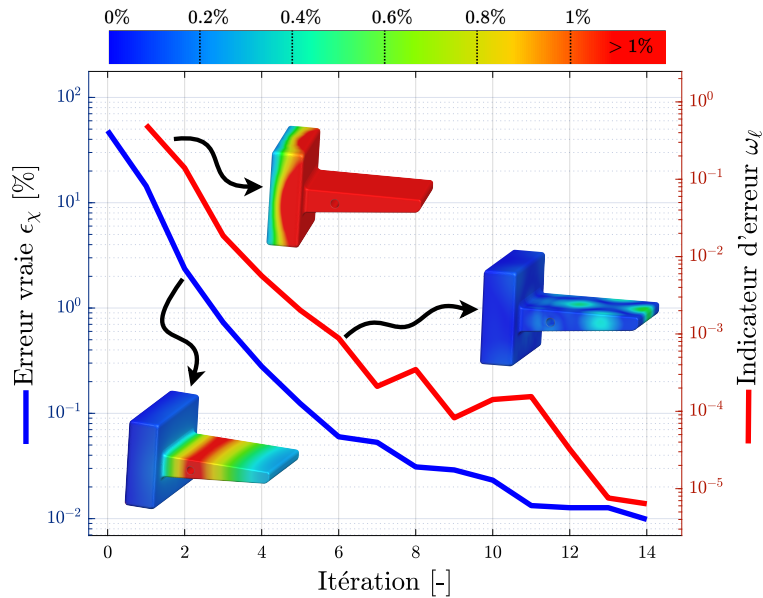


FIGURE 5 – Évolution de l'indicateur d'erreur ω_ℓ et de l'erreur vraie ϵ_χ en fonction des itérations avec illustrations spatiales des niveaux d'erreur relative : $\max_{t \in I} \frac{|\mathbb{T}_\ell(t) - \mathbb{T}_\chi(t)|}{|\mathbb{T}_\chi(t)|}$ (en %).

Il est possible d'utiliser des approximations moins précises de la solution, en augmentant le critère d'arrêt associé à l'indicateur d'erreur ω_ℓ . La stratégie non-incrémentale LATIN-PGD a l'avantage de fournir rapidement une approximation de la solution pour l'ensemble de l'espace et du temps. Cela est particulièrement utile dans les environnements industriels où le temps de calcul est souvent limité. La LATIN-PGD permet ainsi d'obtenir une approximation de la solution dans le temps imparti. Si nécessaire, il est ensuite possible d'améliorer la qualité de l'approximation obtenue en utilisant l'approximation précédente comme solution initiale de l'algorithme LATIN-PGD.

Finalement, la Figure 6 représente la décomposition de la solution en une somme de produits de fonctions spatiales et temporelles. Le premier mode capture principalement la réponse globale de la structure au chargement cyclique associé au débit du liquide de refroidissement, tandis que les modes suivants tiennent compte de comportements locaux au niveau du perçage cylindrique ou de l'extrémité de l'ailette. La réutilisation intelligente des différents modes PGD générés lors des calculs peut être mise à profit pour l'étude de solutions non linéaires de problèmes multiparamétriques. La méthode LATIN-PGD offre en effet un cadre propice à ce type d'étude, la structure même de l'algorithme permettant de redémarrer un calcul à partir de n'importe quelle approximation précédemment convergée proche dans l'espace paramétrique.

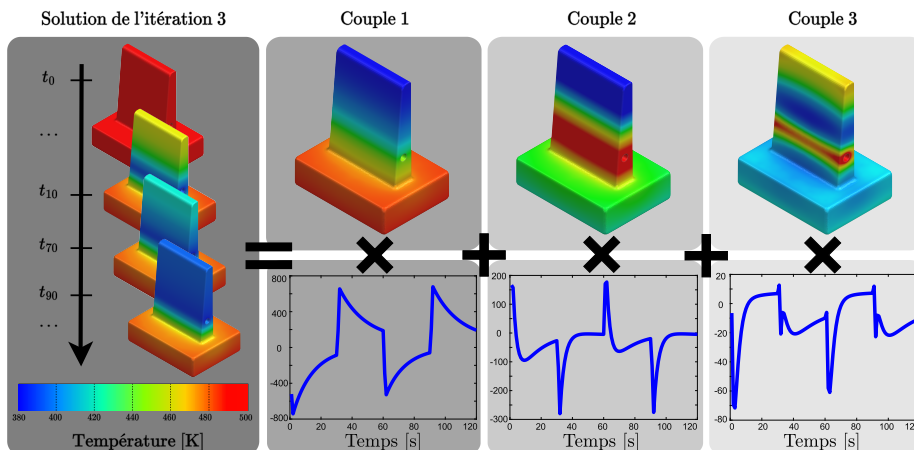


FIGURE 6 – Décomposition en fonctions du temps et de l'espace de la solution obtenue à l'issue de l'itération 3 (possédant 3 couples).

5 Conclusion

Les travaux présentés illustrent la force de l'intégration d'une méthode de réduction de modèles (LATIN-PGD) au cœur du code source d'un logiciel industriel (Simcenter Samcef). Cette intégration, réalisée à l'aide d'une formulation faiblement intrusive de l'algorithme LATIN, permet d'étendre le démonstrateur préexistant [13] aux problèmes thermiques transitoires, mettant en évidence la robustesse de l'approche. L'ensemble des possibilités offertes par le logiciel hôte peuvent être réexploitées, en englobant ainsi tous types d'éléments, tous types de lois de comportement, tous types de de non-linéarités, etc. Le cas-test présenté recense par exemple une loi de comportement non linéaire (fonction de la température) et quatre types de conditions limites dont deux non linéaires. En outre, la possibilité de traiter le problème intégralement dans le logiciel hôte facilite grandement la création ou modification de la géométrie, le maillage, la mise en donnée ainsi que l'analyse des résultats. La mise en œuvre de cas industriels multi-paramétriques, l'étude des performances de l'approche ainsi que son optimisation, font partie des travaux en cours.

6 Remerciements

Nous remercions Siemens Digital Industries Software ainsi que l'Association Nationale de la Recherche et de la Technologie (ANRT) pour le financement de ces travaux.

Références

- [1] P. Benner, S. Grivet-Talocia. *Model Order Reduction : Snapshot-Based Methods and Algorithms, volume 2*, De Gruyter, 2021.
- [2] 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, John Wiley and Sons Ltd, 32-53, 2020.
- [3] A. Chatterjee. *An introduction to the proper orthogonal decomposition*, Current Science, Current Science Association, 808-817, 2021.
- [4] F. Chinesta, A. Leygue, M. Beringhier, L. Tuan Nguyen, J.-C. Grandidier, B. Schrefler, F. Pesavento. *Towards a framework for non-linear thermal models in shell domains*, International Journal of Numerical Methods for Heat & Fluid Flow, 2013.
- [5] F. Chinesta, R. Keunings, A. Leygye. *The Proper Generalized Decomposition for Advanced Numerical Simulations : A Primer*, Springer, 2013.
- [6] F. Chinesta, P. Ladevèze. *Proper generalized decomposition. In : Model Order Reduction : Snapshot-Based Methods and Algorithms* De Gruyter, 2020.
- [7] E. Cueto, D. González, I. Alfaro. *PGD for Non-linear Problems. In : Proper Generalized Decompositions*, Springer International Publishing, Cham, 2016.
- [8] C. Ghnatios, E. Cueto, A. Falco, J.-L. Duval, F. Chinesta. *Spurious-free interpolations for non-intrusive PGD-based parametric solutions : Application to composites forming processes*, International Journal of Material Forming, Springer Paris, 83-95, 2021.
- [9] R. Ibañez, E. Abisset-Chavanne, A. Ammar, D. González, E. Cueto, A. Huerta, J.-L. Duval, F. Chinesta. *A Multidimensional Data-Driven Sparse Identification Technique : The Sparse Proper Generalized Decomposition*, Complexity, Hindawi, 1-11, 2018.
- [10] P. Ladevèze. *Nonlinear Computational Structural Mechanics : New Approaches and Non-Incremental Methods of Calculation*, Springer, New York, 1999.
- [11] Y. Maday, EM. Rønquist. *The Reduced Basis Element Method : Application to a Thermal Fin Problem*, SIAM Journal on Scientific Computing, Society for Industrial and Applied Mathematics Publications, 240-258, 2004.
- [12] G. Rozza. *Reduced Basis Approximation and Error Bounds for Potential Flows in Parametrized Geometries*, Communications in Computational Physics, Global Science Press, 1-48, 2011.
- [13] R. Scanff, D. Néron, P. Ladevèze, P. Barabinot, F. Cugnon, J.-P. Delsemme. *Weakly-invasive LATIN-PGD for solving time-dependent non-linear parametrized problems in solid mechanics*, Computer Methods in Applied Mechanics and Engineering, Elsevier, 114999, 2022.
- [14] X. Zou, M. Conti, P. Díez, F. Auricchio. *A nonintrusive proper generalized decomposition scheme with application in biomechanics*, International Journal for Numerical Methods in Engineering, John Wiley and Sons Ltd, 230-251, 2016.