

Évaluation de l'approche data-driven pour les grandes transformations élastiques

A. Platzer, L. Stainier, A. Leygue

GeM, Ecole Centrale Nantes, {auriane.platzer, laurent.stainier, adrien.leygue}@ec-nantes.fr

Résumé — L'approche data-driven [6] a récemment été étendue aux grandes transformations élastiques pour les treillis, et pour les problèmes 2D au travers d'un exemple [9]. Nous proposons ici une formulation éléments finis générique du problème ainsi qu'une nouvelle méthode de résolution du système non linéaire. Les performances de la méthode sont alors évaluées sur un problème mécanique 2D induisant des champs hétérogènes, pour une base de données contenant uniquement trois états de déformation différents.

Mots clés — data-driven, élasticité non linéaire.

1 Introduction

Les inconnues d'un problème de mécanique quasi-statique sont (i) le champ de déplacement, (ii) le champ de déformation et (iii) le champ de contrainte. Elles sont reliées par deux types d'équations : d'une part, l'équilibre mécanique et la compatibilité des déformations sont universellement valides ; d'autre part, le modèle de comportement dépend uniquement du matériau étudié. L'utilisation d'un modèle de comportement peut présenter plusieurs limitations : de l'identification de ses paramètres à sa formulation mathématique en passant par l'extrapolation de son domaine de validité, des biais de modélisation sont introduits à chaque étape d'un processus phénoménologique complexe. Dans le même temps, les outils expérimentaux pour la mécanique des matériaux n'ont cessé de se perfectionner pour permettre de générer un grand nombre de données [1, 3]. Toutefois, la masse des données ainsi obtenues est souvent perdue une fois les paramètres du modèle identifiés.

Kirchdoerfer & Ortiz ont récemment proposé de reformuler le problème mécanique pour s'affranchir du modèle de comportement et ainsi utiliser directement une base de données matériau, constituée de couples contrainte-déformation [6]. Formulée en élasticité linéaire, cette nouvelle approche de simulation numérique par les données (*data-driven computational mechanics*) consiste à assurer la compatibilité et l'équilibre des états de contrainte et de déformation de la structure, en s'approchant « au plus près » du comportement du matériau, défini par la base de données. Cette approche a depuis conduit à d'autres applications, comme une méthode inverse pour l'identification de la base de données matériau [8], ou bien la simulation de comportements anélastiques [10]. Nguyen & Keip ont également proposé une extension « naturelle » de l'approche data-driven aux grandes transformations élastiques, fournissant une formulation du problème pour les treillis de barre puis une implémentation particulière sur un problème analytique 2D [9].

Dans ce travail, nous proposons une formulation générique du problème data-driven en éléments finis en grandes déformations élastiques (transformations réversibles et indépendantes de l'histoire subie par le matériau). La méthode de résolution du système non linéaire qui en résulte, utilisée dans [9], est améliorée afin de réduire le coût du calcul. Enfin, le contenu de la base de données matériau nécessaire à la simulation d'un problème mécanique 2D induisant des champs hétérogènes est discuté : on démontre ainsi qu'un certain échantillonnage de la richesse des états de déformation vus par la structure doit être présent dans la base de données pour obtenir des résultats cohérents.

2 Méthodes

2.1 Formulation éléments finis du problème data-driven en grandes transformations

L'approche data-driven consiste à minimiser, dans un espace contrainte-déformation (appelé *espace des phases*) et sur l'ensemble des points d'intégration de la structure, la distance entre les états issus d'une base de données matériau d'une part et des états respectant à la fois l'équilibre des contraintes et la compatibilité des déformations d'autre part. Dans le cadre des grandes transformations, Nguyen & Keip ont choisi de formuler le problème sur la configuration non déformée; l'espace des phases est donc l'ensemble des couples de tenseurs (\mathbf{E}, \mathbf{S}) déformation de Green-Lagrange – contrainte de Piola-Kirchhoff II [9]. Le problème est alors formulé comme suit :

$$\text{solution} = \arg \min_{\mathbf{E}, \mathbf{S}} \sum_{e=1}^m \min_i \frac{w_e}{2} [(\mathbf{E}_e - \mathbf{E}_i^*) : \mathbb{C} : (\mathbf{E}_e - \mathbf{E}_i^*) + (\mathbf{S}_e - \mathbf{S}_i^*) : \mathbb{C}^{-1} : (\mathbf{S}_e - \mathbf{S}_i^*)] \quad (1a)$$

sous contraintes

$$\mathbf{F}_e = \mathbf{I} + \nabla_0 \mathbf{u} \quad \text{et} \quad \mathbf{E}_e = \frac{1}{2} (\mathbf{F}_e^T \mathbf{F}_e - \mathbf{I}), \forall e \quad (1b)$$

$$\mathbf{R}_a = \sum_{e=1, e \ni a}^{m_a} w_e \nabla_0 N_a \cdot (\mathbf{S}_e \mathbf{F}_e^T) - \mathbf{T}_a^{\text{ext}} = \mathbf{0}, \forall a \quad (1c)$$

avec

- $(\mathbf{E}_i^*, \mathbf{S}_i^*)$ le $i^{\text{ème}}$ couple de la base de données matériau, \mathbb{C} un tenseur du 4^{ème} ordre définissant la mesure de distance dans l'espace des phases et que l'on suppose symétrique (symétries mineures et majeures), w_e le poids d'intégration et m le nombre total de points d'intégration dans le maillage;
- \mathbf{F}_e le tenseur gradient de la transformation au point d'intégration e , $\nabla_0 \mathbf{u}$ le gradient du champ de déplacement et \mathbf{I} le tenseur identité;
- \mathbf{R}_a le vecteur résidu au nœud a , m_a étant le nombre de points d'intégration de tous les éléments connectés au nœud a , $\mathbf{T}_a^{\text{ext}}$ le vecteur des efforts extérieurs au nœud a , que l'on suppose indépendant de \mathbf{u} , et $\nabla_0 N_a$ le vecteur gradient de la fonction de forme au nœud a [2].

La minimisation (1a) s'effectue d'une part sur les variables continues (\mathbf{E}, \mathbf{S}) , appelés *états mécaniques*, et d'autre part sur des variables discrètes, les états de la base de données matériau $(\mathbf{E}_i^*, \mathbf{S}_i^*)$. On est donc face à un problème complexe de minimisation continue couplée à une optimisation combinatoire qui présente potentiellement de nombreux minima locaux.

2.2 Résolution alternée des problèmes de minimisation continue et discrète

La méthode de résolution la plus simple et initialement proposée dans [9, 6] consiste à résoudre alternativement le problème continu puis le problème combinatoire d'appariement entre les points de la base de données matériau et les états mécaniques. Cette approche de point fixe est facile à mettre en œuvre mais on peut d'ors et déjà anticiper qu'elle ne fournira que des minima locaux. Cette méthode de résolution est appelée Algorithme 1, où on note $(\mathbf{E}_{ie}^*, \mathbf{S}_{ie}^*)$ l'*état matériau* du point d'intégration e , défini comme le couple de la base de données le plus proche de l'état mécanique de e :

$$(\mathbf{E}_{ie}^*, \mathbf{S}_{ie}^*) = \arg \min_i \frac{1}{2} [(\mathbf{E}_e - \mathbf{E}_i^*) : \mathbb{C} : (\mathbf{E}_e - \mathbf{E}_i^*) + (\mathbf{S}_e - \mathbf{S}_i^*) : \mathbb{C}^{-1} : (\mathbf{S}_e - \mathbf{S}_i^*)], \forall e. \quad (2)$$

À états matériau fixés, c-à-d à appariements ie fixés, le problème (1) se résume alors à un problème de minimisation unique sur les variables continues (\mathbf{E}, \mathbf{S}) , sous contraintes d'équilibre et de compatibilité.

2.2.1 Formulation du problème de minimisation continue en grandes transformations

Dans le problème de minimisation continue sous contraintes, la compatibilité des déformations (1b) est – à la manière de l'approche classique – directement injectée dans la définition de la fonction objectif (1a). La contrainte d'équilibre des efforts (1c) est assurée par l'intermédiaire de multiplicateurs de

Algorithme 1 Algorithme général de résolution alternée du problème data-driven.

$k \leftarrow 0$ initialisation des itérations

1. $ie^{(k)}, \forall e \leftarrow$ premier appariement arbitraire des états mécaniques et matériau

2. $(\mathbf{E}_e^{(k+1)}, \mathbf{S}_e^{(k+1)}), \forall e \leftarrow$ résolution de (1) à états matériau fixés : $(\mathbf{E}_i^*, \mathbf{S}_i^*) = (\mathbf{E}_{ie^{(k)}}^*, \mathbf{S}_{ie^{(k)}}^*)$

3. $(\mathbf{E}_{ie^{(k+1)}}^*, \mathbf{S}_{ie^{(k+1)}}^*), \forall e \leftarrow$ mise à jour des états matériau (2)

4. test de convergence : $ie^{(k)} = ie^{(k+1)}, \forall e ?$

— si oui, **fin**.

— si non, $k \leftarrow k + 1$, **retour 2**.

Lagrange η . Le problème se résume alors à écrire la stationnarité du Lagrangien suivant :

$$\Pi = \sum_{e=1}^m \frac{w_e}{2} [\Delta \mathbf{E}_e(\mathbf{u}) : \mathbb{C} : \Delta \mathbf{E}_e(\mathbf{u}) + \Delta \mathbf{S}_e : \mathbb{C}^{-1} : \Delta \mathbf{S}_e] - \sum_{a=1}^n \eta_a \cdot \mathbf{R}_a, \quad (3)$$

où $\Delta \mathbf{E}_e(\mathbf{u}) = \mathbf{E}_e(\mathbf{u}) - \mathbf{E}_{ie}^*$ (respectivement $\Delta \mathbf{S}_e = \mathbf{S}_e - \mathbf{S}_{ie}^*$) est l'écart sur les tenseurs de déformation (respectivement contrainte) mécanique et matériau. Contrairement au cas des petites déformations, la stationnarité du Lagrangien conduit à un système d'équations non linéaires :

$$\frac{\partial \Pi}{\partial \mathbf{S}_e} = 0, \forall e \Rightarrow \quad \mathbf{S}_e(\mathbf{u}, \eta) = \mathbb{C} : (\nabla_0^T \eta \mathbf{F}_e) + \mathbf{S}_{ie}^*, \forall e \quad (4a)$$

$$\frac{\partial \Pi}{\partial \eta_a} = 0, \forall a \Rightarrow \quad \sum_{e=1, e \ni a}^{m_a} w_e [\nabla_0 N_a \cdot (\mathbf{S}_e \mathbf{F}_e^T)] - \mathbf{T}_a^{\text{ext},(e)} = \mathbf{0}, \forall a \quad (4b)$$

$$\frac{\partial \Pi}{\partial \mathbf{u}_a} = 0, \forall a \Rightarrow \quad \sum_{e=1, e \ni a}^{m_a} w_e [\nabla_0 N_a \cdot (\mathbb{C} : (\Delta \mathbf{E}_e) \mathbf{F}_e^T) - \nabla_0 N_a \cdot (\mathbf{S}_e \nabla_0^T \eta)] = \mathbf{0}, \forall a \quad (4c)$$

La résolution du système (4) se résume alors à l'annulation de deux résidus non linéaires et couplés en \mathbf{u} et η , définis ci-après :

$$\mathbf{R}_a^\eta(\mathbf{u}, \eta) = \sum_{e=1, e \ni a}^{m_a} w_e [\nabla_0 N_a \cdot (\mathbb{C} : (\nabla_0^T \eta \mathbf{F}_e) \mathbf{F}_e^T) + \nabla_0 N_a \cdot (\mathbf{S}_{ie}^* \mathbf{F}_e^T)] - \mathbf{T}_a^{\text{ext},(e)} \quad (5a)$$

$$\mathbf{R}_a^u(\mathbf{u}, \eta) = \sum_{e=1, e \ni a}^{m_a} w_e [\nabla_0 N_a \cdot (\mathbb{C} : (\Delta \mathbf{E}_e) \mathbf{F}_e^T) - \nabla_0 N_a \cdot (\mathbb{C} : (\nabla_0^T \eta \mathbf{F}_e) \nabla_0^T \eta) - \nabla_0 N_a \cdot (\mathbf{S}_{ie}^* \nabla_0^T \eta)] \quad (5b)$$

\mathbf{R}_a^η traduit l'application de la contrainte d'équilibre de l'état mécanique et \mathbf{R}_a^u traduit la minimisation de la distance entre l'état mécanique et l'état matériau associé, tout en assurant la compatibilité des déformations. On vérifie aisément que la linéarisation de ces deux quantités sous l'hypothèse des petites perturbations permet de retrouver le système linéaire obtenu dans [6]. On souligne également que même si le problème data-driven est formulé en (\mathbf{E}, \mathbf{S}) , l'Eq. (5) montre qu'il est résolu en (\mathbf{u}, \mathbf{S}) par l'intermédiaire de la compatibilité $\mathbf{F}_e = \mathbf{F}_e(\nabla_0 \mathbf{u})$ et des multiplicateurs de la Lagrange η .

2.2.2 Méthodes de résolution du problème de minimisation continue

Nguyen & Keip ont proposé une méthode de résolution directe du système (5) par un algorithme de Newton–Raphson, rappelé Algorithme 2. Cette méthode assure la compatibilité et l'équilibre des états mécaniques, obtenus avec les états matériau déterminés à l'itération k de l'Algorithme 1, avec une grande précision. Pourtant, avant convergence sur les appariements, c-à-d lorsque les états mécaniques sont encore « loin » des états matériau, il n'est pas nécessaire de résoudre de façon fine les contraintes de compatibilité et d'équilibre.

Nous proposons alors de découpler le système (4) : nous résolvons alternativement l'équilibre (*i.e.* déterminer η et \mathbf{S} pour \mathbf{u} connu) puis la compatibilité et la distance entre états mécaniques et matériau (*i.e.* déterminer \mathbf{u} avec \mathbf{S} et η connus). Cette méthode de résolution, appelée découplée dans la suite et décrite Algorithme 3, consiste à résoudre d'abord un système linéaire en η de taille $n_{\text{ddl}} \times n_{\text{ddl}}$ puis à résoudre un système non linéaire en \mathbf{u} de taille $n_{\text{ddl}} \times n_{\text{ddl}}$.

Algorithme 2 Méthode de résolution directe proposée par [9].

2. $k_{i\grave{e}me}$ itération de l'algorithme 1

• $j_{\grave{e}me}$ itération de Newton–Raphson :

$$\Delta \mathbf{q}_b^{(k,j)} \leftarrow \sum_{b=1}^{n_a} \mathbf{K}_{ab} \cdot \Delta \mathbf{q}_b^{(k,j)} = - [\mathbf{R}_a^u, \mathbf{R}_a^\eta]^T \quad \text{avec } \mathbf{K}_{ab} = \begin{pmatrix} \frac{\partial \mathbf{R}_a^u}{\partial \mathbf{u}_b} & \frac{\partial \mathbf{R}_a^u}{\partial \boldsymbol{\eta}_b} \\ \frac{\partial \mathbf{R}_a^\eta}{\partial \mathbf{u}_b} & \frac{\partial \mathbf{R}_a^\eta}{\partial \boldsymbol{\eta}_b} \end{pmatrix}$$

$$\mathbf{q}^{(k,j+1)} \leftarrow \mathbf{q}^{(k,j)} + \Delta \mathbf{q}^{(k,j)} \quad \text{avec } \mathbf{q} = [\mathbf{u}, \boldsymbol{\eta}]^T$$

3. mise à jour des états matériau (2)

Algorithme 3 Nouvelle méthode de résolution découplée.

2. $k_{i\grave{e}me}$ itération de l'algorithme 1

• $\boldsymbol{\eta}^{(k+1)}, \mathbf{S}^{(k+1)} \leftarrow$ résolution du système linéaire suivant :

$$\begin{cases} \sum_e w_e \left[\nabla_0 N_a \cdot \left(\mathbb{C} : \left(\nabla_0^T \boldsymbol{\eta}^{(k+1)} \mathbf{F}_e^{(k)} \right) \mathbf{F}_e^{T(k)} \right) \right] = \mathbf{T}_a^{\text{ext},(e)} - \sum_e w_e \nabla_0 N_a \cdot \left(\mathbf{S}_{ie^{(k)}}^* \mathbf{F}_e^{T(k)} \right), \forall a \\ \mathbf{S}_e^{(k+1)} (\mathbf{u}^{(k)}, \boldsymbol{\eta}^{(k+1)}) = \mathbb{C} : \left(\nabla_0^T \boldsymbol{\eta}^{(k+1)} \mathbf{F}_e^{(k)} \right) + \mathbf{S}_{ie^{(k)}}^*, \forall e \end{cases}$$

• $\mathbf{u}^{(k+1)} \leftarrow$ résolution du système non linéaire suivant, par un algorithme de Newton-Raphson :

$$\sum_e w_e \left[\nabla_0 N_a \cdot \left(\mathbb{C} : \left(\Delta \mathbf{E}_e^{(k+1)} \right) \mathbf{F}_e^{T(k+1)} \right) \right] = \sum_e w_e \left[\nabla_0 N_a \cdot \left(\mathbf{S}_e^{(k+1)} \nabla_0^T \boldsymbol{\eta}^{(k+1)} \right) \right], \forall a$$

3. mise à jour des états matériau (2)

3 Résultats

Nous évaluons dans cette partie la formulation et la méthode de résolution du problème data-driven proposées sur un problème d'élasticité 2D (contraintes planes). Nous interrogeons la richesse de la base de données matériau pour reproduire fidèlement des résultats obtenus avec un modèle de comportement. Dans cette optique nous travaillons avec des données générées artificiellement à partir d'un modèle de comportement et comparons l'approche classique avec modèle (EF) et l'approche data-driven (DD).

3.1 Génération de la base de données

Les états matériau sont générés à partir d'un modèle de comportement hyperélastique incompressible isotrope néo–Hookéen, sous l'hypothèse des contraintes planes, dont le paramètre est $C_{10} = 0,6$ MPa. Ce modèle est ensuite « oublié » : il n'intervient pas dans la résolution du problème data-driven.

La base de données est composée de couples contrainte-déformation correspondant à trois essais standards et contient l'isotropie du matériau. Dans une première étape, les états matériau sont générés dans la base principale : le tenseur gradient de la transformation matériau s'écrit $\mathbf{F}^* = \lambda \mathbf{e}_1 \otimes \mathbf{e}_1 + \lambda^B \mathbf{e}_2 \otimes \mathbf{e}_2 + \lambda^{-1-B} \mathbf{e}_3 \otimes \mathbf{e}_3$. L'extension λ prend alors N_λ valeurs dans l'intervalle $[1, 2]$ et le facteur de multi-axialité B prend seulement 3 valeurs :

- $B = -0,5$ pour un essai de traction uniaxiale,
- $B = 0$ pour un essai de cisaillement pur,
- $B = 1$ pour un essai de traction équibiaxiale.

Ces trois essais ont été choisis car ils sont communément utilisés pour calibrer les modèles de comportement hyperélastiques des élastomères [5]. La deuxième étape consiste à enrichir artificiellement la base de données pour prendre en compte l'isotropie du matériau : à chaque couple de tenseurs $(\bar{\mathbf{E}}^*, \bar{\mathbf{S}}^*)$ dans ses directions principales, on ajoute également toute l'orbite $(\mathbf{R}\bar{\mathbf{E}}^*\mathbf{R}^T, \mathbf{R}\bar{\mathbf{S}}^*\mathbf{R}^T)$ à la base de données. Ici, \mathbf{R} est une matrice de rotation 2D, dont l'angle θ prend N_θ valeurs dans l'intervalle $[0^\circ, 180^\circ]$. L'orbite de chaque tenseur est donc discrétisée d'un pas $\Delta\theta = 180^\circ / (N_\theta - 1)$.

Les résultats présentés ci-après ont été obtenus pour $N_\lambda = 2000$ et $N_\theta = 181$, soit $N_d = 725\,638$ points dans la base de données, sauf précision contraire.

3.2 Géométrie, maillage et conditions aux limites de la structure étudiée

La structure étudiée est une membrane trouée dont les dimensions sont données sur la Figure 1. Le maillage comporte $m = 1072$ éléments triangulaires linéaires (1 seul point d'intégration), soit $n_{ddl} = 1128$ degrés de liberté. La structure est encastree à sa base ($\mathbf{u}(X, Y = 0) = \mathbf{0}$) et chargée en déplacement sur la partie haute : $\mathbf{u}(X, Y = h) = 0,03\mathbf{e}_X + 0,005\mathbf{e}_Y, \forall X$.

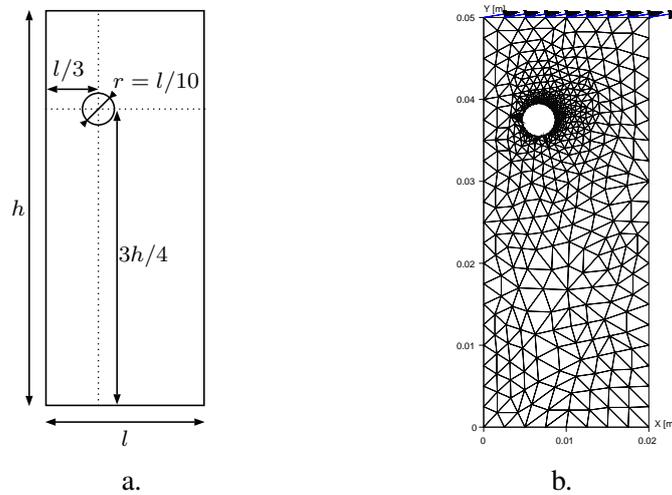


FIGURE 1 – Géométrie, maillage et chargement de la structure étudiée. a. Géométrie de la structure. b. Maillage et déplacement imposé.

3.3 Résultats obtenus par la nouvelle méthode de résolution

Les résultats de la simulation DD, obtenus avec la nouvelle méthode de résolution (Algorithme 3), sont comparés avec ceux provenant d'une simulation éléments finis, où le modèle de comportement est celui utilisé pour la génération de la base de données. Sur la Figure 2a, on observe que la déformée prédite par la simulation DD est cohérente avec la déformée EF. Néanmoins, les erreurs sur le champ de déplacement sont plus importantes autour du trou, comme le montre la Figure 2b. On observe également un écart non négligeable sur les efforts résiduels à l'endroit du déplacement imposé.

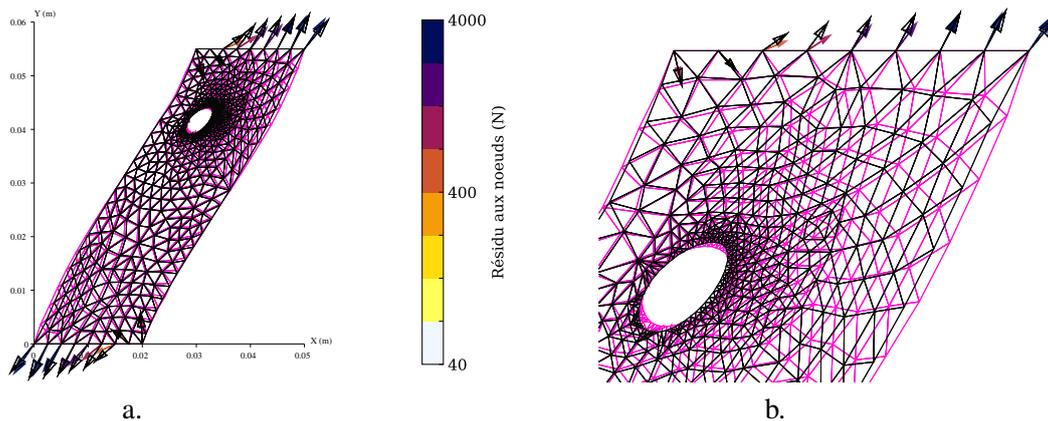


FIGURE 2 – a. Comparaison des déformées de la structure (simulation EF en noir et simulation DD en rose) et des efforts résiduels (DD : flèches colorées, EF : flèches creuses noires). b. Zoom autour du trou et de la partie haute de la membrane.

3.4 Convergence comparée des deux algorithmes

Nous nous assurons de la robustesse de la nouvelle méthode de résolution découplée en comparant ses propriétés de convergence à la méthode proposée dans [9]. La Figure 3a démontre que l'évolution de

la valeur de la fonction objectif (1a) en fonction du nombre N_d de points dans la base de données est la même, quelle que soit la méthode utilisée. Sur la Figure 3b on montre même que le nombre d'itérations nécessaires à la convergence de l'algorithme général data-driven à chaque pas de temps est parfois plus favorable pour la méthode découplée.

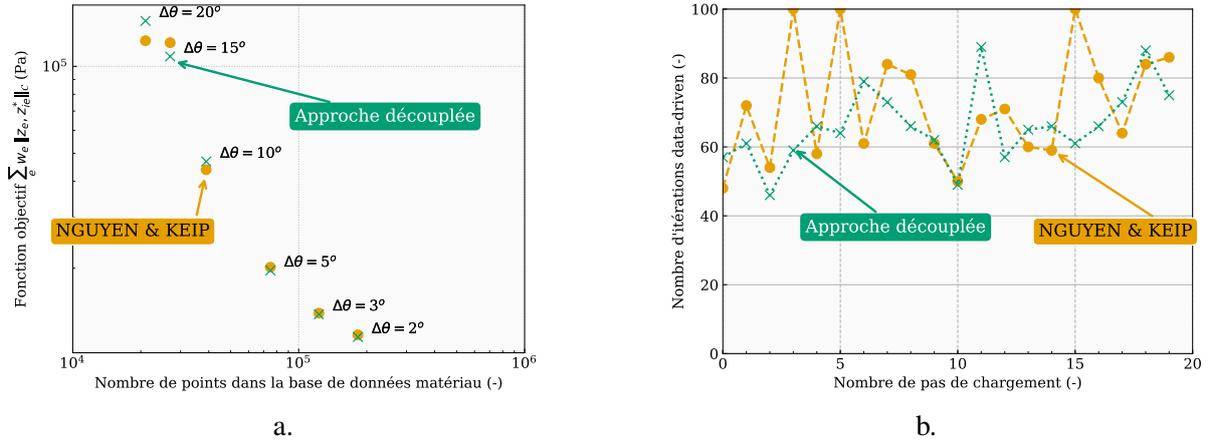


FIGURE 3 – Convergence comparée de la méthode de résolution directe (Algorithme 2) et découplée (Algorithme 3). a. Valeur de la fonction objectif (1a) en fonction du nombre de points dans la base de données matériau à $N_\lambda = 1000$ fixé et N_θ variable. b. Nombre d'itérations data-driven (Algorithme 1) en fonction du pas de chargement pour les deux méthodes, à $N_\lambda = 2000$ et $N_\theta = 181$ fixés, soit avec $N_d = 725\,638$ points dans la base de données matériau.

4 Discussion

La fonction objectif de l'algorithme data-driven est la distance entre les états mécaniques et les états matériau. La Figure 3a montre que lorsqu'on augmente la densité de points dans la base de données, les états mécaniques se rapprochent bien des états matériau. Pour autant, les résultats de la simulation DD qui sont les états mécaniques, ne sont pas en accord avec le solution de référence EF (Figure 2). Nous attribuons donc ces erreurs au manque de richesse de la base de données d'une part, et à la faiblesse intrinsèque de l'algorithme général de résolution alternée (Algorithme 1) d'autre part.

4.1 Richesse de la base de données

La base de données matériau utilisée pour la simulation DD ne comporte que 3 types d'états de déformation : la traction uniaxiale, équi-biaxiale et le cisaillement pur. Pourtant, le chargement et la géométrie de la structure induisent des états de déformation plus complexes, d'après la simulation EF. Sur la Figure 4, on trace le premier invariant I_1 de $\mathbf{C} = \mathbf{F}^T \mathbf{F}$ en fonction du second I_2 , à tous les points d'intégration du maillage, pour la simulation DD d'une part, et la simulation EF d'autre part. On montre ainsi que les états mécaniques de déformation de la simulation DD se concentrent sur des états matériau de déformation essentiellement en traction uniaxiale et sur quelques états de cisaillement pur. En comparant avec la répartition des états de déformation sur la simulation EF, on remarque que les états intermédiaires situés entre la courbe de traction uniaxiale et la courbe de cisaillement pur ne sont pas reproduits, notamment à faible extension.

Le problème data-driven est donc bien résolu en déformation : les états mécaniques sont très proches des états matériau ; mais puisque la base de données ne contient pas les états de déformation solution, l'écart entre les simulations DD et EF est important. C'est ce que l'on observe sur la Figure 5 qui présente l'écart relatif sur les normes de Von Mises des tenseurs de déformation et de contrainte :

$$\varepsilon_E = \left| \frac{E_{VM}^{DD} - E_{VM}^{EF}}{E_{VM}^{EF}} \right| \quad \text{et} \quad \varepsilon_S = \left| \frac{S_{VM}^{DD} - S_{VM}^{EF}}{S_{VM}^{EF}} \right| \quad (6a)$$

$$\text{avec} \quad E_{VM} = \sqrt{\frac{2}{3} \mathbf{E}_{dev} : \mathbf{E}_{dev}} \quad \text{et} \quad S_{VM} = \sqrt{\frac{3}{2} \mathbf{S}_{dev} : \mathbf{S}_{dev}} \quad (6b)$$

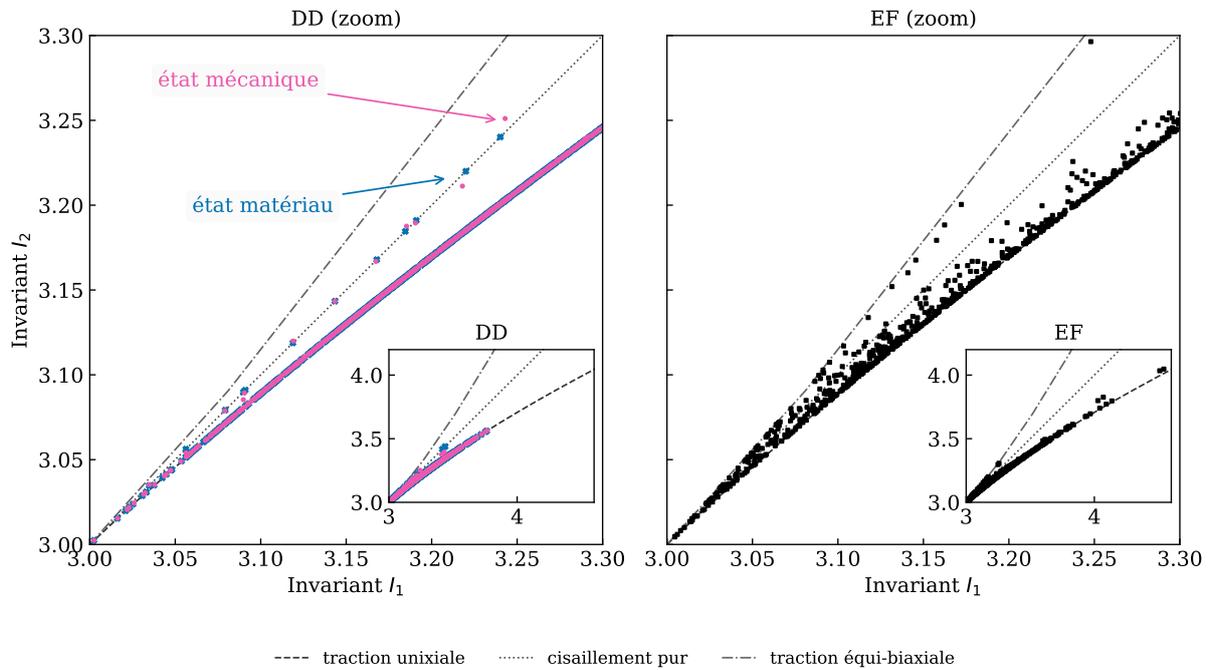


FIGURE 4 – États de déformation en chaque point d’intégration du maillage de la simulation DD à gauche et de la simulation EF à droite. A gauche, les états mécaniques (solution du problème) sont représentés en cercles roses et les états matériau associés (selon (2)) en croix bleues. Sur tous les graphes, le lieu des états pour les 3 essais standards sont représentés en lignes pointillées.

On obtient des erreurs assez importantes (majoritairement supérieures à 5%) et relativement plus importantes pour les déformations. Ce dernier point s’explique par la valeur du tenseur \mathbb{C} : dans les simulations présentées nous avons choisi un tenseur d’élasticité constant, dont le module est 100 fois supérieur au paramètre matériau C_{10} . De cette manière, on accorde plus de poids à la distance entre les états de déformation ($\Delta E : \mathbb{C} : \Delta E$) qu’entre les états de contrainte ($\Delta S : \mathbb{C}^{-1} : \Delta S$), de par la forme de la norme choisie (2) : les états de déformation mécanique sont très proches des états de déformation matériau (Figure 4a) et donc loin de la solution EF (Figure 5).

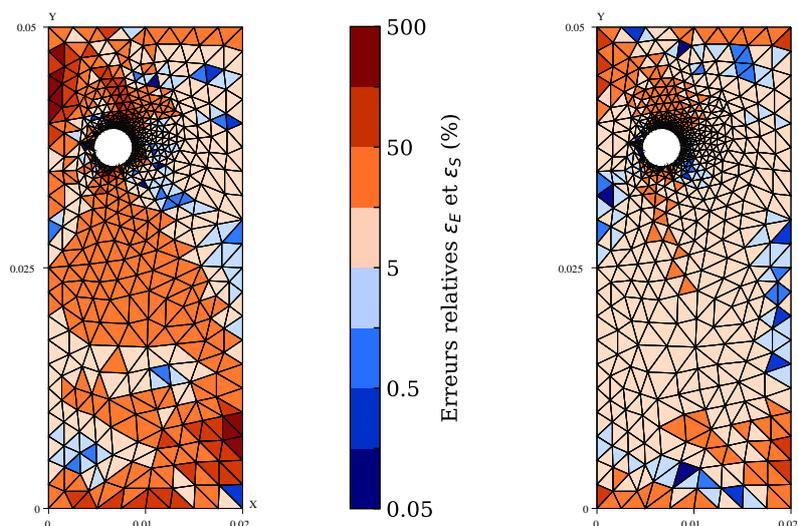


FIGURE 5 – Erreurs relatives sur les déformations (à gauche) et sur les contraintes (à droite) au sens de Von Mises, avec échelle de couleur logarithmique : les zones rouges pales sont comprises entre 5 et 15,81 %, les zones bleues sont inférieures à 5%.

4.2 Performance de l'algorithme

Outre la richesse de la base de données, l'imprécision de la simulation DD provient également de la faiblesse de l'algorithme de résolution. En effet, le problème à minimiser comporte un grand nombre de minima locaux, de par le caractère discret de la base données. Des résultats préliminaires obtenus avec une base de données constituée exclusivement des résultats issus de la simulation EF, indiquent que la méthode de résolution alternée ne parvient pas à trouver le minimum global. En effet, même lorsque l'appariement initial (point **1**. de l'Algorithme 1) du premier pas de chargement est imposé par la solution plutôt qu'arbitraire, la solution DD dérive un peu plus à chaque pas de chargement de la solution EF, pourtant présente dans la base de données. Une amélioration de l'algorithme de résolution générale apparaît donc nécessaire, comme envisagé par Kirchdoerfer et Ortiz [7] pour ignorer les points aberrants de bases de données bruitées, ou bien l'utilisation d'une toute autre classe d'algorithmes, comme proposé par Kanno [4].

5 Conclusion

En conclusion, ce travail fournit une formulation éléments finis générique (2D ou 3D) du problème data-driven en grandes transformations élastiques. Il apporte également une nouvelle méthode de résolution moins coûteuse pour le problème de minimisation non linéaire qui apparaît, du fait de la non linéarité de la relation de compatibilité. Enfin, cette étude démontre la nécessité d'une base de données riche en termes d'états de déformation : les 3 essais standards qui permettent d'identifier les modèles de comportement ne suffisent plus pour réaliser une simulation data-driven.

En perspective, il convient donc d'évaluer l'approche data-driven en grandes transformations avec des données plus riches : soit provenant d'essais expérimentaux hétérogènes dépouillés au moyen d'une méthode inverse de type [8], soit issus de simulations numériques sur des microstructures composites pour lesquelles les modèles de comportement des constituants sont connus et considérés comme fiables. Enfin, l'algorithme alterné initialement proposé doit être exploré plus avant afin de mieux résoudre le problème combinatoire des appariements.

Références

- [1] S. Avril, M. Bonnet, A.S. Bretelle et al. *Overview of identification methods of mechanical parameters based on full-field measurements*. *Experimental Mechanics*, (48) 381-402, 2008.
- [2] J. Bonet, R.D. Wood. *Nonlinear continuum mechanics for finite element analysis*, Cambridge University Press, 243-245, 2008.
- [3] J. Y. Buffiere, E. Maire, J. Adrien, J.P. Masse, E. Boller. *In situ experiments with X ray tomography : an attractive tool for experimental mechanics*. *Experimental Mechanics*, (50) 289-305, 2010.
- [4] Y. Kanno. *Mixed-integer programming formulation of a data-driven solver in computational elasticity*. arXiv preprint : 1810.04394v1, 2018.
- [5] R. W. Ogden. *Large deformation isotropic elasticity – on the correlation of theory and experiment for incompressible rubberlike solids*. *Proc. R. Soc. Lond. A*, (326) 565-584, 1972.
- [6] T. Kirchdoerfer, M. Ortiz. *Data-driven computational mechanics*. *Computer Methods in Applied Mechanics and Engineering*, (304) 81-101, 2016.
- [7] T. Kirchdoerfer, M. Ortiz. *Data-driven computing with noisy material data sets*. *Computer Methods in Applied Mechanics and Engineering*, (326) 622-641, 2017.
- [8] A. Leygue, M. Coret, J. Réthoré, L. Stainier, E. Verron. *Data-based derivation of material response*. *Computer Methods in Applied Mechanics and Engineering*, (331) 184-196, 2018.
- [9] L.T. Nguyen, M.A. Keip. *A data-driven approach to nonlinear elasticity*. *Computers and Structures*, (194) 97-115, 2018.
- [10] R. Eggersmann, T. Kirchdoerfer, S. Reese, L. Stainier, M. Ortiz. *Model-free data-driven inelasticity*. arXiv preprint : 1808.10859v2, 2018.