

# Optimisation topologique par le principe de moindre action : une implémentation PyTorch

O. Jeanne<sup>1,2</sup>, T. Cherrière<sup>2</sup>, A. Daby-Seesaram<sup>3,4</sup>

<sup>1</sup> ENS Paris-Saclay, DER Génie Mécanique, Gif-sur-Yvette, France, [octave.jeanne@ens-paris-saclay.fr](mailto:octave.jeanne@ens-paris-saclay.fr)

<sup>2</sup> GeePs, CentraleSupélec, Sorbonne Université, CNRS, Gif-sur-Yvette, France, [theodore.cherriere@centralesupelec.fr](mailto:theodore.cherriere@centralesupelec.fr)

<sup>3</sup> LMS, École polytechnique, Institut Polytechnique de Paris, CNRS, Palaiseau, France

<sup>4</sup> ENSTA-Paris, Institut Polytechnique de Paris, Palaiseau, France [alexandre.daby-seesaram@ensta-paris.fr](mailto:alexandre.daby-seesaram@ensta-paris.fr)

---

## Abstract —

En général, l'optimisation de structure consiste en une suite de résolutions du problème physique, éventuellement non-linéaire, qui converge vers un optimum local. Or, il n'est pas toujours utile de résoudre la physique de manière exacte : pour gagner en temps de calcul, on peut résoudre les deux problèmes, équations physiques et optimisation structurale, simultanément. Pour ce faire, on se base sur le principe de moindre action, et on propose une implémentation par réseaux de neurones interprétables. La méthode est appliquée au design d'une inductance en génie électrique.

**Mots clés** — Optimisation topologique, Réseaux de neurones interprétables, Méthode des éléments finis, Principe de moindre action.

---

## 1 Introduction

L'optimisation topologique est un champ de recherche actif. Initialement développée en mécanique des structures [2], elle s'est depuis étendue à de nombreux domaines comme le design de composants magnétiques en ingénierie électrique [5]. La majorité du temps de calcul est alors occupée par la résolution du problème physique non-linéaire par la méthode des éléments finis. Une idée (qui n'est pas nouvelle) est de réduire ce temps de calcul en ne résolvant pas exactement la physique au cours du processus d'optimisation. On traite alors les grandeurs physiques et de design comme des variables d'un problème d'optimisation plus large, la physique ne devant être vérifiée qu'à convergence [6].

Ce formalisme s'adapte naturellement aux problèmes physiques qui satisfont le principe de moindre action, i.e., qui proviennent de la minimisation d'une fonctionnelle d'énergie. Cette propriété thermodynamique peut être exploitée dans les problèmes d'optimisation topologique [7]. En effet, certaines fonctions objectifs utiles dans le cadre d'un problème d'optimisation de structures peuvent être minimisées de concert avec l'énergie, comme la compliance, dans des problèmes linéaires aussi bien que non-linéaires [1]. Cette quantité est d'une grande utilité physique, et constitue la grandeur d'intérêt d'une large part des problèmes d'optimisation topologique.

Par ailleurs, résoudre la physique dans le problème d'optimisation est compatible avec l'utilisation de réseaux de neurones : reformuler un problème physique comme la minimisation d'une fonction coût est le fondement des "Physics Informed Neural Networks" (PINNs, [12]). Ces modèles présentent une grande expressivité; en revanche, ces architectures de réseaux de neurones ne sont généralement pas interprétables. Le cadre "Hierarchical Deep-learning Neural Networks" (HideNN, [9, 13]) propose un compromis entre expressivité et interprétabilité permettant de conserver le bénéfice des récentes avancées liées à l'apprentissage profond (différentiation automatique, compatibilité native avec le calcul sur GPUs, par exemple). L'idée est de reproduire l'interpolation par éléments finis avec des réseaux de neurones parcimonieux. L'entraînement de ces modèles porte non-seulement sur la solution physique, mais aussi sur des paramètres supplémentaires qui peuvent être liés au maillage (position des noeuds) ou à la répartition de matière, ce qui n'est pas nativement le cas dans la méthode des éléments finis classique. Le cadre HideNN rend ainsi possible la résolution simultanée du problème physique et d'optimisation topologique. Enfin, les bibliothèques de différenciation automatique comme PyTorch [11] rendent la mise en œuvre de ce formalisme relativement aisée.

Cet article s’organise autour des parties suivantes. Tout d’abord, le formalisme des réseaux de neurones interprétables pour les éléments finis [9] est rappelé, et son implémentation est validée pour une simulation d’une équation de Poisson <sup>1</sup>. Ce formalisme est ensuite étendue à l’optimisation topologique d’une inductance en magnéto-statique.

## 2 Formalisme

### 2.1 Discrétisation du champ solution

Afin de résoudre numériquement une équation aux dérivées partielles (EDPs), il est en général nécessaire de la discrétiser, et, donc, de proposer une interpolation de la solution continue dans un espace de dimension finie. La méthode des éléments finis est un choix classique pour définir cette interpolation. Dans ce cadre, un champ  $u(x)$  est projeté sur une base de  $N$  fonctions de forme  $\{N_i(x)\}_{i \in \llbracket 1, N \rrbracket}$  tel que

$$u(x) \simeq u_h(x) = \sum_{i=1}^N u_i N_i(x) \quad (1)$$

avec  $\{u_i\}_{i \in \llbracket 1, N \rrbracket}$  les paramètres associés à cette interpolation. Cette méthode propose un espace d’interpolation relativement rigide mais offre une grande interprétabilité puisque les  $\{u_i\}_{i \in \llbracket 1, N \rrbracket}$  correspondent à la valeur du champ interpolé aux différents nœuds du maillage. Cela permet, entre autres, de fixer aisément des conditions de Dirichlet lorsque l’on résout une EDP, et de proposer des solveurs optimisés.

À l’opposé du spectre, les méthodes basées sur des réseaux de neurones proposent une interpolation offrant une grande expressivité au coût d’une interprétabilité réduite. Ainsi, l’interpolation d’un champ  $u$  avec un réseau de neurones à  $M + 1$  couches s’écrit comme

$$u_h = \sigma \left( \sum_{k=1}^{N_M} w_{1k}^{(M)} y_k^{(M)} + b_1^{(M+1)} \right) \quad (2)$$

où  $N_M$  correspond au nombre de neurones de la couche  $M$ ,  $b_j^{(i)}$  au biais associé au  $i$ -ème neurone de la  $j$ -ème couche,  $\sigma$  à la fonction d’activation choisie,  $w_{ij}^{(l)}$  au poids synaptique de la couche  $l$  reliant le neurone  $j$  (de la couche précédente) au neurone  $i$  (de la couche actuelle) et

$$y_k^{(l)} = \sigma \left( \sum_{j=1}^{N_{l-1}} w_{kj}^{(l)} y_j^{(l-1)} + b_k^{(l)} \right) \quad (3)$$

à la valeur du  $k$ -ième neurone de la couche  $l$ . L’architecture HiDeNN reproduit l’interpolation éléments finis, en contraignant les biais et les poids du réseau, ce qui le rend interprétable. Une vision schématique de cette idée est proposée en Figure 1.

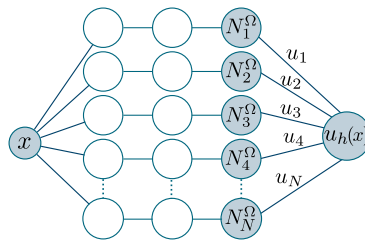


Figure 1: Illustration de l’interpolation par réseau HiDeNN

L’espace d’interpolation  $\text{Vect}(\{N_i(x)\}_{i \in \llbracket 1, N \rrbracket})$  et les opérateurs d’intégration issus du problème physique sont généralement fixes lorsqu’on utilise la méthode des éléments finis. Le formalisme HiDeNN propose ainsi d’accroître l’expressivité sans perdre en interprétabilité en intégrant les paramètres des fonctions de forme et de la physique au problème d’optimisation. Par ailleurs, HiDeNN peut bénéficier de l’accélération matérielle GPU en étant implémenté à l’aide des bibliothèques de machine learning.

<sup>1</sup>Les calculs s’appuient sur la bibliothèque neurom-optim (basée sur le code NeuROM [4]), accessible [sur ce dépôt](https://octave-jeanne.github.io/NeuROM_optim_demo/). Une démonstration de son utilisation permettant de reproduire les figures est disponible au lien suivant : [https://octave-jeanne.github.io/NeuROM\\_optim\\_demo/](https://octave-jeanne.github.io/NeuROM_optim_demo/).

## 2.2 Formulation typique d'un problème HiDeNN

À l'instar de la méthode des éléments finis, le cadre HiDeNN exploite le fait que la solution d'un système physique correspond souvent à un état d'équilibre minimisant une certaine fonctionnelle d'énergie. L'étape d'entraînement consiste donc à naviguer dans l'espace des paramètres (par exemple constitué des valeurs et coordonnées nodales de la solution), pour atteindre le minimum global de la fonctionnelle d'énergie définie par la physique du problème. Ce faisant, l'entraînement des réseaux HiDeNN est en mesure de joindre la résolution du problème physique avec l'optimisation du maillage sous-jacent. Dans le cas d'un problème statique conservatif, on définit la fonctionnelle d'énergie

$$\mathcal{L} := E_p^{\text{int}} - W^{\text{ext}}, \quad (4)$$

avec  $E_p^{\text{int}}$  l'énergie potentielle stockée et  $W^{\text{ext}}$  le travail forces généralisées externes, qu'on peut minimiser avec un optimiseur tel qu'Adam [8] ou L-BFGS [10] par exemple. Si la fonctionnelle à minimiser pour résoudre le problème d'optimisation topologique coïncide avec la fonction coût physique, il suffit d'inclure les variables de design dans le problème de minimisation. Le problème à résoudre s'écrit

$$\left( \{u_i\}_{i \in \llbracket 1, N \rrbracket}, \{\rho_i\}_{i \in \llbracket 1, p \rrbracket} \right) = \operatorname{argmin} \mathcal{L}, \quad (5)$$

avec  $\left( \{u_i\}_{i \in \llbracket 1, N \rrbracket} \right)$  les valeurs nodales du champ et  $\{\rho_i\}_{i \in \llbracket 1, p \rrbracket}$  les autres paramètres relatifs au maillage, à la répartition de matière, etc. On peut alors résoudre les problèmes physiques, d'optimisation topologique et d'optimisation de maillage simultanément. Le problème physique est détaillé en Section 3, puis l'optimisation topologique qui repose sur cette double minimisation est présentée en Section 4. On ne traite pas de l'optimisation du maillage dans cet article.

## 3 Formulation magnétostatique 2D à flux imposé

On considère un domaine  $\Omega$  en statique, et en l'absence de courants sources. De l'équation de Maxwell-Ampère et de la loi matériau  $\underline{b} = \mu \underline{h}$ , on tire l'équation de la magnétostatique 3D:

$$\operatorname{rot} \left( \frac{1}{\mu(x, y, z)} \operatorname{rot} \underline{a} \right) = 0, \quad (6)$$

avec  $\mu \in L^\infty(\Omega)$  la perméabilité magnétique (éventuellement dépendante de l'induction  $\underline{b}$ ),  $\underline{a}$  le potentiel vecteur magnétique tel que  $\underline{b} = \operatorname{rot} \underline{a}$ ; le théorème de Stokes fournit alors une relation entre flux et potentiel vecteur  $\iint_S \underline{b} \cdot d\underline{s} = \oint_{\partial S} \underline{a} \cdot d\underline{l}$ . La solution de (6) n'est pas forcément identiquement nulle, puisqu'on peut imposer un flux  $\phi$  par des conditions de Dirichlet. On peut montrer que dans le cas où  $\underline{b}$  est dans le plan  $(x, y)$ , alors (6) peut se réécrire comme une équation de Laplace sur la composante orthogonale  $z$ :

$$\operatorname{div} \left( \frac{1}{\mu(x, y)} \operatorname{grad} a_z \right) = 0. \quad (7)$$

Imposer un flux au travers d'une surface située entre deux points  $p^+ = (x^+, y^+)$  et  $p^- = (x^-, y^-)$  et de côté unitaire dans l'axe  $z$  revient alors à fixer la différence entre  $a_z(p^+)$  et  $a_z(p^-)$ . On peut alors décomposer  $a_z = a_0 + a_d$ , avec  $a_0 \in H_0^1(\Omega) := \{a \in H^1(\Omega), a|_{\partial\Omega \cup \{p^+\} \cup \{p^-\}} = 0\}$  et  $a_d \in H_d(\Omega) := \{a \in H^1(\Omega), a|_{\partial\Omega} = 0, a(\{p^+\}) = \phi/2, a(\{p^-\}) = -\phi/2\}$ . La formulation faible de (7) consiste alors à trouver le champ  $a_0 \in H_0^1(\Omega)$ , qui pour un  $a_d \in H_d(\Omega)$  vérifiant les conditions de Dirichlet, satisfait l'égalité

$$\int_{\Omega} \operatorname{grad} v \cdot \frac{1}{\mu(x, y)} \operatorname{grad} (a_0 + a_d) \, dx dy = 0, \quad \forall v \in H_0^1(\Omega). \quad (8)$$

Résoudre (8) est équivalent à résoudre le problème de minimisation sous contrainte suivant [1] :

$$\min_{a_z \in H_d(\Omega)} J(a_z) := \int_{\Omega} w(x, y, \operatorname{rot}_2(a_z)) \, dx dy, \quad (9)$$

avec  $\underline{\text{rot}}_2(a_z) = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \underline{\text{grad}}(a_z) = \underline{\text{rot}} \underline{a}$  pour un problème plan, et  $w$  une densité d'énergie magnétique caractéristique du matériau, strictement convexe selon  $\underline{\text{rot}}_2(a_z)$ . Pour un matériau magnétique doux, on a

$$w(\underline{b}) = \int_{\underline{\beta}=0}^{\underline{b}} \underline{h}(\underline{\beta}) \cdot d\underline{\beta}, \quad (10)$$

l'intégrale ne dépendant pas du chemin suivi, et si de plus le matériau est linéaire, la densité d'énergie associée est une forme quadratique  $w(\underline{b}) = \frac{|\underline{b}|^2}{2\mu}$ . La fonctionnelle  $J$  définie dans (9) peut-être directement implémentée, discrétisée et minimisée dans le formalisme HiDeNN. Les contraintes de Dirichlet inhérentes à l'espace  $H_d(\Omega)$  sont imposées simplement en fixant les valeurs des variables sur  $\partial\Omega$ ,  $p^+$  et  $p^-$ . Un algorithme d'optimisation peut alors ajuster les autres degrés de libertés pour résoudre (9).

À titre d'exemple de validation de l'implémentation, la Figure 2 montre une comparaison entre les résultats issus de la minimisation d'une fonctionnelle d'énergie dans le formalisme HiDeNN (algorithme L-BFGS) et d'une implémentation éléments finis sur NGSolve dans le cadre d'un problème de Poisson pour des matériaux linéaires, avec ajout de terme source homogène. Les maillages étant identiques, les erreurs sur les calculs flottants simple précision expliquent peut-être la faible différence.

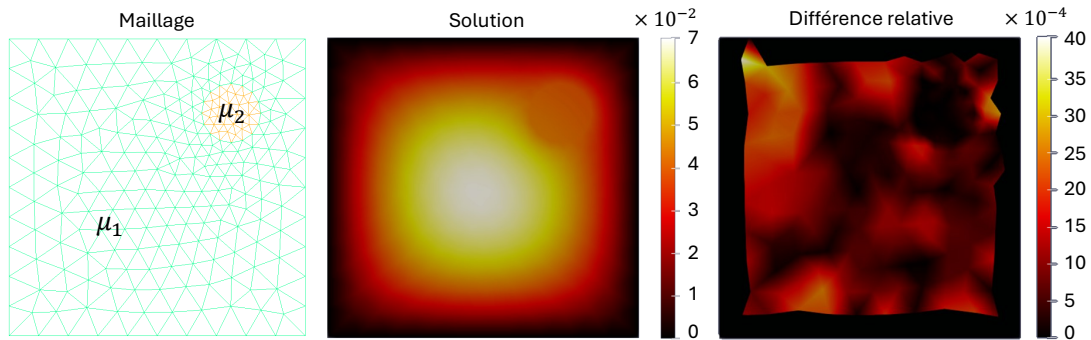


Figure 2: Solution d'une équation de Poisson (centre) sur un même maillage (gauche), et différence relative (droite) entre une implémentation éléments finis et HiDeNN.

## 4 Optimisation topologique d'une inductance

Le problème de minimisation (9) peut être élargi à des variables de design dans la zone à optimiser  $\Omega_D \subset \Omega$ , puisque l'objectif est également de minimiser l'énergie de magnétisation. Dans la méthode d'optimisation topologique à densité [2, 5], on considère comme variable d'optimisation un champ de densité qui permet d'interpoler les propriétés matériaux. Le problème s'écrit alors comme une double-minimisation, pour lesquelles il existe de nombreux algorithmes de résolution :

$$\min_{\rho \in L^\infty(\Omega_D, [0,1])} \min_{a_z \in H_d(\Omega)} J(a_z, \rho), \quad (11)$$

avec

$$J(a_z, \rho) := \int_{\Omega \setminus \Omega_D} w_0(\underline{\text{rot}}(a_z)) dx dy + \int_{\Omega_D} w_D(\underline{\text{rot}}(a_z), \rho) dx dy. \quad (12)$$

Les matériaux en présence sont l'air de perméabilité  $\mu_0 = 4\pi \times 10^{-7} \text{H/m}$  et du fer supposé linéaire de perméabilité relative initiale  $\mu_r = 1000$ , si bien qu'on exprime la densité d'énergie par la formule suivante (d'autres choix d'interpolation sont possibles)

$$w_D(\underline{b}, \rho) = \rho \frac{|\underline{b}|^2}{2\mu_0\mu_r} + (1 - \rho) \frac{|\underline{b}|^2}{2\mu_0}. \quad (13)$$

Ainsi,  $w_d(|\underline{b}|, \rho = 0)$  correspond à la densité d'énergie de l'air, et  $w_d(|\underline{b}|, \rho = 1)$  à celle du matériau ferromagnétique. Imposer l'appartenance de  $\rho$  dans l'intervalle  $[0, 1]$  sans imposer de contrainte supplémentaire dans l'optimisation peut se faire en introduisant une variable auxiliaire  $\hat{\rho}$  et une fonction  $\hat{\sigma} : \mathbb{R} \rightarrow [0, 1]$

strictement croissante, comme la sigmoïde  $\rho = \hat{\sigma}(\hat{\rho}) = \frac{1}{1+\exp(-\hat{\rho})}$ . Notons que l'écriture du problème (12) ne changerait pas en présence de non-linéarités dues à la saturation du matériau magnétique ; par ailleurs, les densités d'énergie resteraient convexes selon [b]. Ce n'est pas nécessairement le cas de  $J$ , même si des algorithmes d'optimisation peuvent tout de même être appliqués.

Le cas test choisi est une inductance représentée Figure 3, qui suit la physique décrite à la Section 3. Il s'agit d'une bobine enroulée autour d'un noyau de fer doux, qui constitue un des composants de base du génie électrique pour des applications de filtrage.

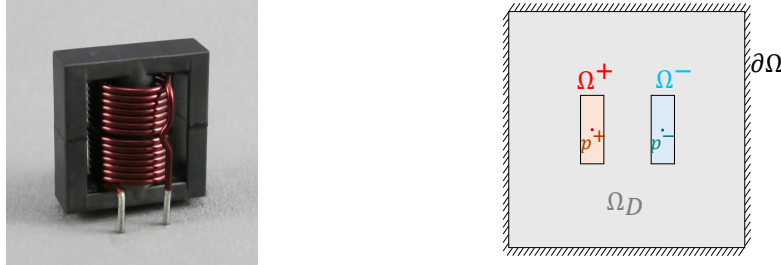


Figure 3: Inductance à noyau magnétique en E (gauche, crédits : [info.triadmagnetics.com](http://info.triadmagnetics.com)), et modélisation 2D en coupe du domaine de design (droite).

Le flux magnétique entre  $p^+$  et  $p^-$  étant fixé, on souhaite minimiser l'énergie magnétique dépensée pour magnétiser le circuit. Cette énergie est maximale lorsque la perméabilité du circuit est faible ( $\rho = 0$ ), et minimale lorsque la perméabilité est grande ( $\rho = 1$ ). Pour éviter une solution triviale et limiter le coût du circuit magnétique, on impose une contrainte de volume du circuit ferromagnétique à l'aide d'un multiplicateur de Lagrange  $\lambda$  arbitraire, pour finalement obtenir la fonction objectif à minimiser

$$J_L(a_z, \rho) = \int_{\Omega \setminus \Omega_D} w_0(\text{rot}(a_z)) dx dy + \int_{\Omega_D} w_D(\text{rot}(a_z), \rho) dx dy + \frac{\lambda}{|\Omega_D|} \int_{\Omega_D} \rho dx dy. \quad (14)$$

C'est cette fonctionnelle qui est en pratique exprimée dans le formalisme HideNN-FEM et minimisée selon  $a_z$  et  $\rho$  simultanément. On utilise pour cela l'algorithme L-BFGS sur 2000 epochs au maximum, et un critère d'arrêt sur la stagnation relative de l'énergie de  $10^{-9}$  (les autres hyperparamètres de Pytorch sont ceux par défaut). D'autres optimiseurs et critères d'arrêt sont possibles, cf [3]. Un front de Pareto tracé à la Figure 4 est obtenu pour différentes valeurs de  $\lambda$ .

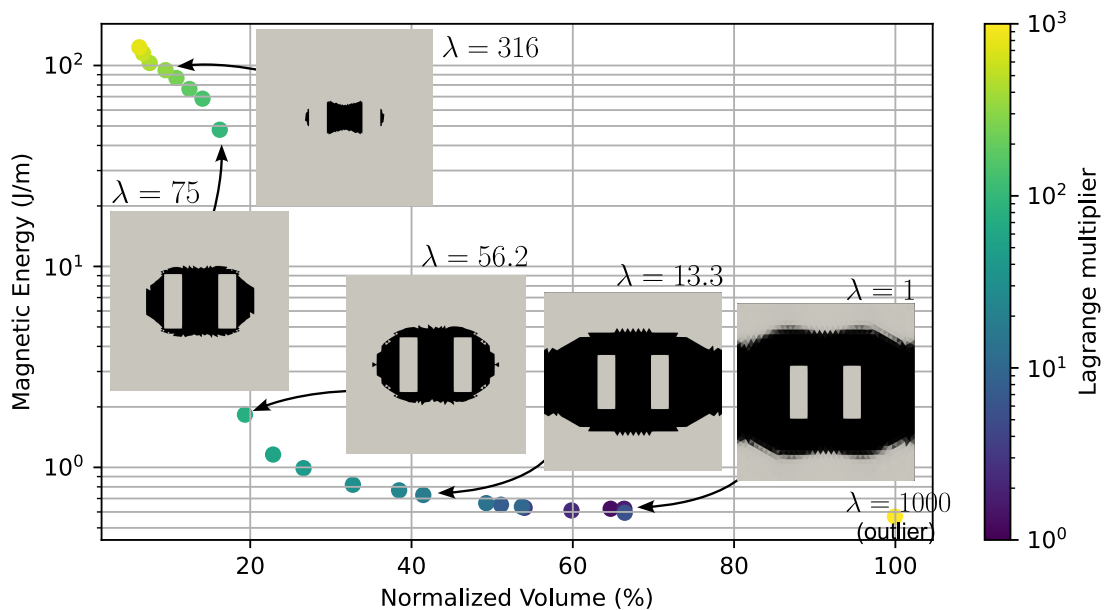


Figure 4: Front de Pareto énergie magnétique / volume et structures optimisées (foncé : matériau ferromagnétique, clair : matériau non-magnétique).

Les géométries optimisées pour des  $\lambda$  intermédiaires ressemblent à des structures classiques d'inductances, dont un exemple est représenté Figure 3. Nous travaillons sur la prise en compte de la saturation

magnétique : on s’attend alors à ce que les structures ferromagnétiques fines qui concentrent le flux magnétique soient saturées, et donc qu’elles s’élargissent au cours de l’optimisation.

## 5 Conclusion

La méthode des éléments finis a été implémentée sous le formalisme HideNN, qui permet davantage de flexibilité. Un des potentiels de cette méthode est son application à l’optimisation topologique, effectuée simultanément à la résolution de la physique.

Afin d’exploiter le plein potentiel de la méthode, de futurs travaux sont nécessaires pour l’étendre aux matériaux non-linéaires et tirer partie de l’accélération matérielle par GPU. Ces deux points ne devrait pas poser trop de difficultés grâce à l’utilisation de la bibliothèque Pytorch, qui gère à la fois la dérivation automatique et son portage sur GPU, ce qui permettra d’améliorer les performances de la méthode, encore peu compétitives en termes de temps de calcul.

Par ailleurs, une difficulté inhérente à cette approche est sa limitation à des physiques et fonctions objectifs issues de problèmes de minimisation compatibles. Un cas plus général serait la minimisation sous contrainte et la recherche de points selles, pour lesquels les bibliothèques d’apprentissage automatique n’offre que peu d’algorithmes sur étagère et constitue un axe de travail pour de futures recherches.

## References

- [1] G. Allaire, T. Cherière, T. Gauthey, M. H. Hassan, X. Mininger. A remark on self-adjoint problems in the optimization of non-linear models. *Journal of Optimization Theory and Applications*, 208(3), fév. 2026. doi [10.1007/s10957-025-02928-6](https://doi.org/10.1007/s10957-025-02928-6).
- [2] M. P. Bendsøe, N. Kikuchi. Generating optimal topologies in structural design using a homogenization method. *Computer Methods in Applied Mechanics and Engineering*, 71(2):197–224, 1988. doi [10.1016/0045-7825\(88\)90086-2](https://doi.org/10.1016/0045-7825(88)90086-2).
- [3] A. Daby-Seesaram, K. Škardová, M. Genet. Finite Element Neural Network Interpolation: Part II - Hybridisation with the Proper Generalised Decomposition for non-linear surrogate modelling. *Computational Mechanics*, 2025. doi [10.1007/s00466-025-02676-4](https://doi.org/10.1007/s00466-025-02676-4).
- [4] A. Daby-Seesaram, K. Škardová, M. Genet. NeuROM. Zenodo, oct. 2025, doi [10.5281/zenodo.13772740](https://doi.org/10.5281/zenodo.13772740).
- [5] D. N. Dyck, D. A. Lowther. Automated design of magnetic devices by optimizing material distribution. *IEEE Transactions on Magnetics*, 32(3 PART 2):1188–1192, 1996. doi [10.1109/20.497456](https://doi.org/10.1109/20.497456).
- [6] R. T. Haftka. Simultaneous analysis and design. *AIAA Journal*, 23(7):1099–1103, juil. 1985. doi [10.2514/3.9043](https://doi.org/10.2514/3.9043).
- [7] D. R. Jantos, K. Hackl, P. Junker. An accurate and fast regularization approach to thermodynamic topology optimization. *International Journal for Numerical Methods in Engineering*, 117(9):991–1017, 2019. doi [10.1002/nme.5988](https://doi.org/10.1002/nme.5988).
- [8] D. P. Kingma, J. Ba. Adam: A Method for Stochastic Optimization. *CoRR*, déc. 2014.
- [9] Y. Lu, H. Li, L. Zhang, C. Park, S. Mojumder, S. Knapik, Z. Sang, S. Tang, D. W. Apley, G. J. Wagner, W. K. Liu. Convolution Hierarchical Deep-learning Neural Networks (C-HiDeNN): finite elements, isogeometric analysis, tensor decomposition, and beyond. *Computational Mechanics*, 72(2):333–362, 2023. doi [10.1007/s00466-023-02336-5](https://doi.org/10.1007/s00466-023-02336-5).
- [10] J. Nocedal. Updating quasi-newton matrices with limited storage. *Mathematics of Computation*, 35(151): 773–782, 1980. doi [10.1090/s0025-5718-1980-0572855-7](https://doi.org/10.1090/s0025-5718-1980-0572855-7).
- [11] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, S. Chintala. PyTorch: An imperative style, high-performance deep learning library. *Advances in Neural Information Processing Systems*, vol. 32, 2019.
- [12] M. Raissi, P. Perdikaris, G. Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019. doi <https://doi.org/10.1016/j.jcp.2018.10.045>.
- [13] K. Škardová, A. Daby-Seesaram, M. Genet. Finite element neural network interpolation: Part I—interpretable and adaptive discretization for solving PDEs. *Computational Mechanics*, août 2025. doi [10.1007/s00466-025-02677-3](https://doi.org/10.1007/s00466-025-02677-3).