Cadre méthodologique pour assurer la spécification d'un simulateur par apprentissage machine profond en vue de sa validation

Christophe Denis

Cadre méthodologique pour assurer la spécification d'un simulateur par apprentissage machine profond en vue de sa validation

Christophe DENIS

Sorbonne Université, LIP6, UMMISCO
Université Panthéon-Sorbonne, IHPST
Saclay INdustrial Collaborative Laboratory for Artificial Intelligence Research (SINCLAIR)
9 novembre 2023
Christophe.Denis@lip6.fr

Les codes de simulation numérique industriels sont basés le plus souvent sur une modélisation hypothético-déductive permettant d'obtenir des avancées significatives sur la prédiction et la compréhension de phénomène. Les techniques d'apprentissage machine profond rendent possible la mise au point de simulateurs sur des phénomènes pour lesquels on ne dispose pas de modèle théorique satisfaisant. Cependant, contrairement à l'approche hypothético-déductive, les simulateurs par apprentissage machine sont développés empiriquement sans être bordés par un cadre formel de spécification et de validation pénalisant leurs acceptabilités opérationnelles. Le cadre méthodologique que nous proposons s'inspire de travaux portant sur la spécification de simulations multi-agent, d'une part, et se base sur une clarification épistémologique de l'apprentissage machine, d'autre part. Cette méthodologie consiste à définir les spécifications formelles du simulateur par apprentissage machine, son modèle sous-jacent, pour ensuite établir une démarche rigoureuse de validation

1 Contexte et motivation industrielle

Le Groupe EDF exploite des matériels et des ouvrages à longue durée de vie comme des barrages hydroélectriques et des centrales nucléaires et mène historiquement au sein de sa R&D une forte activité de simulation numérique dans différents domaines scientifiques, allant de la mécanique des structures à la mécanique des fluides en passant l'électromagnétisme. Ces simulations ont pour objectif de faciliter la prise de décision industrielle sur des quesions de sûreté, de maintenance et d'optimisation de ses moyens de production électrique. A titre d'exemple, le code de simulation des écoulements à surface libre TELEMAC [[Her07]], co-développée par la R&D d'EDF, est utilisé dans le but d'estimer la hauteur d'eau de fleuve afin de dimensionner les digues de protection des centrales nucléaires ou pour décider de l'emplacement d'une hydrolienne afin d'en optimiser sa production. Ce code de simulation comme une grande majorité

d'autres codes industriels se base sur une modélisation hypothético-déductive, qui a permis d'obtenir des avancées significatives sur la prédiction et la compréhension de phénomènes. Le principe de cette modélisation est de formaliser le phénomène d'intérêt sous la forme d'un système d'équations mathématiques exprimées tout d'abord sur des supports continus puis discrétisées sous la forme d'un schéma numérique spatio-temporel. Une méthodologie éprouvée de vérification, de validation et de quantification d'incertitudes est mise en place à EDF R&D pour accréditer les résultats de ses codes de simulation.

La modélisation hypothético-déductive n'est pas toujours la panacée pour questionner un phénomène. En effet, il n'est pas toujours possible de trouver le modèle mathématique d'un phénomène mal connu dans l'état de connaissance scientifique actuel, d'une part, et la simulation de ce modèle peut produire des prédictions peu précises ou nécessiter des temps de calcul ou de consommation énergétique jugés trop importantes, d'autre part. Les techniques d'apprentissage machine profond rendent possible la mise au point de simulateurs sur des phénomènes pour lesquels on ne dispose pas de modèle théorique satisfaisant. Ainsi, de nombreuses disciplines scientifiques computationnelles s'orientent vers l'apprentissage machine profond pour mettre en œuvre des simulateurs sans une formalisation préalable d'un modèle, par exemple pour la simulation de la turbulence en mécanique des fluides. Contrairement à l'approche hypothético-déductive, les simulateurs par apprentissage machine sont développés empiriquement sans être bordés par un cadre formel de spécification et de validation pénalisant leurs acceptabilités opérationnelles, et ce même si on peut vérifier le comportement empirique du simulateur en utilisant des techniques d'interprétation *post hoc*. Il peut être tentant faute de mieux d'adapter pour l'apprentissage machine profond la méthodologie éprouvée de spécification et de validation utilisée pour l'approche hypothético-déductive. Cette solution ne se justifie pas puisque la spécification et la mise au point d'une simulation par apprentissage machine sont différentes de celles mises en œuvre pour une approche hypothético-déductive.

Nous proposons dans cette contribution un cadre méthodologique pour formaliser les spécifications d'un simulateur basée sur de l'apprentissage machine profond pour établir un cadre rigoureux de validation. Ce travail de recherche, mêlant épistémologie de la modélisation et informatique théorique, a pour objectif comme mentionné dans [GS21] de "*réconcilier l'abondance des techniques de vérification de programmes classiques et l'absence de garanties sur les réseaux de neurones pour permettre aux logiciels critiques de conserver leur haut niveau de confiance*". Notre contribution est organisée comme suit. Nous présentons tout d'abord les différences épistémologiques entre une modélisation hypothético-déductive et une modélisation par apprentissage machine. Nous argumentons ensuite sur la nécessité de définir un cadre spécifique de validation de simulateurs par apprentissage, en adaptant à l'apprentissage machine la Théorie de la Modélisation et de la Simulationt pour border epistémologiquement l'utilisation de solutions logicielles de spécifications formelles.

2 Modélisation hypothético-déductive

L'acquisition de la connaissance sur un phénomène nécessite le plus souvent la mise en place d'un objet médiateur que l'on appelle modèle, pour prendre recul

par rapport aux observations. Un modèle peut être une expérience de pensée comme celle de Galilée sur la chute des corps remettant en cause la loi aristotélicienne pour révéler le rôle joué par l'air. Galilée argumente également que l'on ne peut comprendre l'univers *"si l'on ne s'applique d'abord à en comprendre la langue et à connaître les caractères avec lesquels il est écrit. Il est écrit dans la langue mathématique et ses caractères sont des triangles, des cercles et autres figures géométriques, sans le moyen desquels il est humainement impossible d'en comprendre un mot. Sans eux, c'est une errance vaine dans un labyrinthe obscur"*. Initiée par Galilée et Descartes, la modélisation hypothético-déductive a permis d'obtenir des avancées significatives sur la prédiction et la compréhension de phénomènes, non nécessairement physiques d'ailleurs, ce que Wigner qualifiera d'efficacité déraisonnable. Le principe de cette modélisation est de prendre de la distance par rapport aux observations obtenues sur le monde sensible, en formalisant le phénomène sous la forme d'un système d'équations mathématiques qui sont le plus souvent exprimées sur des supports continus. Nous représentons le modèle par une fonction f, comme présenté à la figure 1.

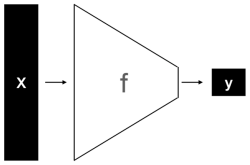

Figure 1: Représentation fonctionnelle d'un modèle

Dans le cadre de l'approche hypothético-déductive,

- la fonction f est spécifiée par le choix des équations mathématiques supposées gouverner le phénomène d'intérêt ;
- le nombre et la nature des données d'entrée X ainsi que la valeur de sortie y sont spécifiés par ces mêmes équations mathématiques.

On peut par exemple choisir comme modèle mathématique les équations de Saint-Venant pour étudier le comportement des écoulements à surface libre. L'évaluation numérique $f(X)$ nécessite le développement d'un simulateur. Pour évaluer numériquement le comportement des écoulements à surface libre, le code de simulation Telemac, co-développé à EDF R&D, construit un schéma spatio-temporel discrétisé pour résoudre les équations mathématiques de Saint-Venant. L'acceptabilité opérationnelle du simulateur nécessite de confronter ces évaluations avec des observables du phénomène. Il est tout d'abord nécessaire de vérifier que le simulateur évalue précisément les trajectoires spécifiées par le modèle. Cette vérification est nécessaire mais non suffisante puisque l'on peut simuler des trajectoires très proches des spécifications du modèle et pour autant éloignées du comportement observable du phénomène. Sur le plan méthodologique, une démarche éprouvée de vérification, de validation et de quantification d'incertitudes a été proposée par [Obe10] :

- l'étape de vérification permet de contrôler la programmation du simulateur spécifiée par la fonction f;
- l'étape de validation évalue la capacité de la fonction f à reproduire les observations du phénomène, ce qui revient à mettre à l'épreuve les spécifications de la fonction f ;
- l'étape de quantification des incertitudes permet d'estimer des intervalles de confiance par rapport aux évaluations de la fonction f, pour tenir compte d'un manque de connaissance de certains comportements du phénomène et des incertitudes sur les paramètres d'entrée.

Cette démarche de vérification et de validation est basée sur l'hypothèse d'une séparation des causes à l'origine de l'écart entre les prédictions et les observations. Cette hypothèse est en cohérence avec les processus de conception et de développement d'un simulateur basée sur une modélisation hypothético-déductive étant donné que le développement du simulateur est toujours encadré par le choix préalable d'un modèle mathématique, qui peut être à son tour remis en cause durant l'étape de validation.

La modélisation hypothético-déductive n'est pas toujours la panacée pour questionner un phénomène. En effet, il n'est pas toujours possible de trouver un modèle mathématique suffisamment précis pour décrire un phénomène mal connu dans l'état de connaissance scientifique actuel, d'une part, et la simulation de ce modèle peut produire des prédictions peu précises ou nécessiter des temps de calcul ou de consommation énergétique jugés trop importantes, d'autre part. Ainsi, de nombreuses disciplines scientifiques computationnelles s'orientent vers l'apprentissage machine profond pour mettre en oeuvre des simulateurs sans une formalisation préalable d'un modèle.

3 Simulation par apprentissage machine profond

Les techniques d'apprentissage machine profond rendent possible la mise au point de simulateurs sur des phénomènes pour lesquels on ne dispose pas de modèle satisfaisant. En reprenant la représentation fonctionnelle de la figure 1, le simulateur prédit une valeur y en fonction de paramètres d'entrée X.

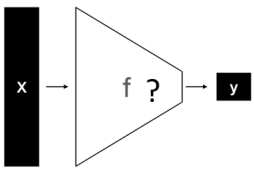

Figure 2: Représentation fonctionnelle d'un simulateur par apprentissage machine

La mise au point du simulateur ne découle pas dans la majorité des cas d'une spécification de la fonction f et des données d'entrée comme dans le cadre de la modélisation hypothético-déductive. Cette spécification s'établit implicitement lors de la mise au point du simulateur. La conception du simulateur suit une

méthodologie empirique fondée sur des connaissances en sciences des données dont la figure 3 représente une synthèse visuelle. Des hypothèses successives sont émises sur l'architecture et les paramètres de la méthode d'apprentissage machine ainsi que sur la représentation des données d'entrée. Ces hypothèses sont confirmées ou non au sein d'une boucle de rétroaction selon l'évolution de la mesure de la métrique d'erreur.

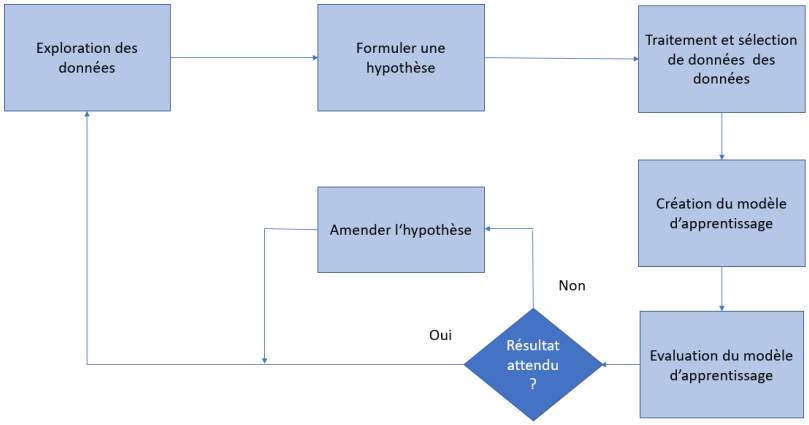

Figure 3: Synthèse visuelle de la méthodologie empirique de mise au point d'un simulateur par apprentissage machine

Ainsi, au fil des itérations i de cette boucle de rétroaction, une succession de fonctions f^i et de représentations des données d'entrée X^i est produit jusqu'à ce qu'un seuil sur la métrique d'erreur soit obtenue avec une méthode d'optimisation. Il est montré dans [NPS21] que les techniques d'optimisation des réseaux de neurones profonds introduit une rupture avec le phénomène cible, pouvant conduire à des spécifications de problèmes différentes de celui du cahier des charges. Ce glissement de spécification s'illustre dans l'exemple bien connu d'un projet de reconnaissance automatique d'un loup ou d'un husky dans des photographies. L'utilisation d'une méthode d'interprétabilité post-hoc a montré que la spécification du simulateur obtenu était différente du cahier des charges car détectant ou non la présence de neige sur les photos. Pour se prémunir en particulier de ce glissement, nous souhaitons mettre en place un cadre formel de spécification pour appliquer systématiquement une méthodologie de vérification et de validation, comme c'est la cas pour les simulateurs basés sur une approche hypothético-déductive.

4 Cadre méthodologique pour la spécification de simulateurs basés sur de l'apprentissage machine profond

Le cadre méthodologique que nous proposons s'inspire des travaux menés par [Que12] pour spécifier des simulations multi-agent, d'une part, et se base sur la clarification épistémologique de l'apprentissage machine proposée par [DV22],

d'autre part. Cette méthodologie consiste à définir les spécifications formelles du simulateur par apprentissage machine, son modèle sous-jacent, afin de mettre en place ensuite en place une démarche rigoureuse de validation. Le terme modèle est par nature polysémique et peut prêter à confusion puisque que ce terme n'a pas la même signification en science des données et en épistémologie de la simulation. Il n'existe pas de distinction en science des données entre le simulateur (dénommé confusément par le terme modèle) qui s'exécute sur un ordinateur, et ses spécifications. Le terme modèle est considéré ici comme un objet médiateur entre un observateur et un phénomène tel que caractérisé par Marvin Minsky "*Pour un observateur B, un objet A* est un modèle d'un objet A, dans la mesure où B peut utiliser A* pour répondre à des questions qui l'intéressent au sujet de A*" [Min68]. Le tableau 1 présente une synthèse du cadre méthodologique de spécification et de validation pour un simulateur construit selon une approche de modélisation hypothético-déductive ou par apprentissage machine profond.

Approches de modélisation	
Hypothético-déductive	Apprentissage machine
Conception du simulateur (au sens de la TMS)	
Modèle ↓ Simulateur	Simulateur ↑ Modèle à formaliser
Vérification et Validation (V&V)	
Méthodologie existante et éprouvée	L'absence de formalisation du modèle ne permet pas de valider rigoureusement le simulateur ⇒ L'objectif est de formaliser le modèle sous-jacent du simulateur pour mettre en place une démarche rigoureuse de validation.

Table 1: Synthèse du cadre méthodologique de spécification.

La caractérisation de Minsky ne peut pas être utilisé pour formaliser les spécifications car elle distingue pas le simulateur du modèle et ne prend pas en compte la méthodologie empirique de mise au point du simulateur. Par contre, il est possible de définir un cadre épsitémologique pertinent en utilisant les concepts issus de la Théorie de la Modélisation et de la Simulation (TMS) proposée dans les années 1970 par Bernard P. Ziegler [ZPK00]. Cette théorie définit les quatre concepts suivants que nous adaptons à l'apprentissage machine profond :

1. Système cible : il s'agit du phénomène d'intérêt pour lequel on cherche à mettre en place un simulateur pour prédire un comportement ou une propriété, comme la hauteur d'eau d'un fleuvre, à partir d'un certain nombre de mesures, qui peuvent être complétées par des données synthétiques.

2. Modèle : il spécifie des relations dynamiques entre des entrées et des sorties. Ces relations dynamiques sont définies par des équations mathé-

matiques dans le cadre d'une approche hypothético-déductive ou par une juxtaposition de fonction de transfert dans le cadre d'une approche par apprentissage machine profond.

3. Simulateur : il produit des trajectoires à partir du comportement d'entrées-sorties par le modèle défini explicitement dans le cadre de l'approche hypoptético-déductive ou implicitement pour l'approche par apprentissage machine profond.

4. Cadre expérimental : il permet de mettre en relation le modèle, le simulateur et le système source dans un cadre d'usage. Il s'agit d'un élément important pour la formalisation de la spécification du simulateur. Le cadre expérimental définit l'ensemble des trajectoires d'états admissibiles en entrée.

Ce cadre permet de border épistémologiquement l'utilisation de solutions logicielles dédiées à la spécification formelle de simulateurs par apprentissage machine profond, comme le logiciel CAMUS traduisant le simulateur en formules logiques [[GSCCS20]]. Enfin, en utilisant les niveaux de spécification proposés par [Kli85], nous souhaitons quantifier la spécification minimale du simulateur à partir de son cahier des charges, d'une part, et de quantifier l'apport de mécanismes d'interprétabilité *post-hoc* afin de le comparer avec le niveau souhaité par le cahier des charges. Il est enfin nécessaire d'associer à ce cadre un travail de fond en informatique théorique pour proposer des nouveaux modèles de calcul pour l'apprentissage machine dépassant la thèse de Church-Turing [Sha22].

Conclusion et perspectives

Les techniques d'apprentissage machine profond permettent de concevoir des simulateurs performants sur des phénomènes ne disposant pas d'un modèle théorique satisfaisant. Cette absence de spécification formelle empêche de mettre en place une démarche rigoureuse de validation. En effet, les paramètres d'un simulateur par apprentissage machine profond s'ajustent dans une boucle de rétroacation visant à réduire une métrique d'erreur. L'optimisation des paramètres est agnostique par rapport au phénomène d'intérêt pouvant induire un glissement implicite de spécification par rapport à celle souhaitée dans le cahier des charges. Ce glissement de spécification peut être détecté en appliquant des techniques d'interprétabilité *post-hoc* lors d'analyses de dysfonctionnements du simulateur. Cette stratégie n'est pas acceptable pour des applications industrielles suffisamment critiques. Cette contribution constitue le fondement épistémologique d'une démarche systématique de validation ne reposant pas sur l'hypothèse peu crédible et non utile d'une compréhension fine des mécanismes internes du simulateur. Il s'agit d'expliciter les spécifications formelles du simulateur, son modèle sous-jacent, en adaptant pour l'apprentissage machine profond la théorie de la Modélisation et de la Simulation, permettant de border l'utilisation d'une solution logicielle de spécification formelle de simulateurs basées sur de l'apprentissage machine profond. Ce cadre méthodologie est en cours de mise à l'épreuve sur un cas d'usage prédisant un phénomène physique en mécanique des fluides et sur la production d'un résumé de texte avec l'agent conversationnel ChatGPT. Enfin, nous poursuivons un travail de

clarification épistémologique pour définir la notion d'algorithme d'apprentissage machine associé au simulateur en nous basant sur des nouveaux modèles de calcul en informatique théorique.

References

[DV22] Christophe Denis and Franck Varenne. Interprétabilité et explicabilité de phénomènes prédits par de l'apprentissage machine. *Revue Ouverte d'Intelligence Artificielle*, 3(3-4):287–310, 2022.

[GS21] Julien Girard-Satabin. *Verification and validation of Machine Learning techniques.* PhD thesis, Université Paris-Saclay, 2021.

[GSCCS20] Julien Girard-Satabin, Guillaume Charpiat, Zakaria Chihani, and Marc Schoenauer. A framework to build formal specifications for deep perception systems using simulators. In *24th European Conference on Artificial Intelligence (ECAI)*, 2020.

[Her07] Jean-Michel Hervouet. *Hydrodynamics of Free Surface Flows: Modelling With the Finite Element Method.* Wiley, 05 2007.

[Kli85] George J. Klir. *Architecture of Systems Complexity.* Sauders, New York, 1985.

[Min68] Marvin Minsky. Matter, mind, and models. pages 425–432, 1968. From Minsky (ed), Semantic Information Processing.

[NPS21] Domenico Napoletani, Marco Panza, and Daniele Struppa. The agnostic structure of data science methods. *Lato Sensu: Revue de la Société de Philosophie des Sciences*, 8:44–57, 2021.

[Obe10] William L. Oberkampf. *Verification and validation in scientific computing/ by William L. Oberkampf and Christopher J. Roy.* Cambridge University Press, Cambridge, 2010.

[Que12] Raphael Duboz; Bruno Bonté ; Gauthier Quesnel. Vers une spécification des modèles de simulation de systèmes complexes. *Studia Informatica Universalis*, 10:7–37, 2012.

[Sha22] Oron Shagir. *The Nature of Physical Computation.* Oxford University Press, 2022.

[ZPK00] Bernard P. Zeigler, Herbert Praehofer, and Tag Gon Kim. *Theory of Modeling and Simulation—Integrating Discrete Event and Continuous Complex Dynamic Systems.* Academic Press, Inc., San Diego, CA, 2nd edition, 2000.