

---

## *Contributions les plus significatives et faits marquants de l'activité de recherche depuis octobre 2016*

---

Dans ce document, je me suis focalisée sur quatre types de travaux récents

1. la conception d'algorithmes distribués
2. l'étude du lien entre l'apprentissage automatique et l'aide à la décision
3. les propriétés structurelles sur les graphes
4. Modèles d'apprentissage et modèle de calculs

### **Algorithmes distribués.**

Je me suis consacrée à la conception d'algorithmes distribués capables de fonctionner en présence de pannes transitoires et de pannes byzantines. Ces deux types de fautes se distinguent par leur nature et leur impact sur le système. Les pannes transitoires, d'une part, peuvent affecter l'ensemble du système, impactant potentiellement tous les nœuds. Toutefois, elles sont de nature temporaire et finissent par disparaître au cours de l'exécution. Le paradigme classique pour traiter ce type de défaut est celui de l'**auto-stabilisation** [19], qui permet à un algorithme de retrouver un fonctionnement correct à partir d'une configuration arbitraire, sans intervention extérieure. Cependant, bien que ces fautes puissent être perturbatrices, elles cessent après un certain temps d'exécution, ce qui permet au système de se rétablir progressivement.

Les **pannes byzantines**, d'autre part, sont fondamentalement différentes car elles ne disparaissent pas avec le temps. Elles concernent exclusivement certains nœuds défectueux et peuvent provoquer un comportement arbitraire ou malveillant de ces derniers. Contrairement aux fautes transitoires, les pannes byzantines sont persistantes : elles peuvent ne jamais cesser tout au long de l'exécution et doivent être prises en compte en permanence dans la conception des algorithmes. Ainsi, nous avons conçu des algorithmes tolérant un ensemble plus large de pannes, dépassant les approches classiques d'auto-stabilisation pour intégrer des mécanismes de résistance aux fautes byzantines.

Un **système réparti** peut être modélisé par un graphe  $G = (V, E)$ , où  $V$  représente l'ensemble des processeurs, et  $E$  désigne les liens de communication entre les processeurs pouvant échanger des informations directement. Chaque processeur ou nœud exécute localement un algorithme, décrit à l'aide de variables partagées et d'actions. Une **configuration** correspond à l'état global du système à un instant donné. Elle est défini comme le produit cartésien des valeurs de toutes les variables associées à chaque nœud du réseau. Chaque nœud gère ses propres variables et échange des informations avec d'autres processeurs via des messages.

Les **actions** sont formulées sous la forme  $\langle \text{règle gardée} \rangle \rightarrow \langle \text{instructions} \rangle$ . Une **règle gardée** est un prédictat défini sur les variables accessibles au processeur. Les instructions associées ne sont exécutées que si l'évaluation de cette règle gardée retourne une valeur vraie.

Un **système de transitions** est défini par un couple  $S = (C, \rightarrow)$ , où  $C$  représente l'ensemble des configurations possibles du système et  $\rightarrow$  est une relation binaire sur  $C$ , décrivant les transitions autorisées entre les configurations.

Une **exécution** de  $S$  est une séquence maximale de configurations, qu'elle soit finie ou infinie, notée  $E = (\gamma_0, \gamma_1, \dots, \gamma_i, \dots)$ , telle que, pour tout  $i \geq 0$ ,  $\gamma_i \rightarrow \gamma_{i+1}$ . En effet, pour chaque paire  $(t, t+1)$  de configurations consécutives, il existe un ensemble non vide de nœuds  $V_t \subseteq V$  tel que : (i) tous les nœuds de  $V_t$  sont activés à  $t$  ; (ii) tous les nœuds  $u \in V \setminus V_t$  conservent le même état local à  $t$  et  $t+1$  ; (iii) l'état local de tous les nœuds  $u \in V_t$  à  $t+1$  est le résultat de son état local à  $t$  et de l'exécution d'une règle pour laquelle  $u$  était activé (on dit que  $u$  effectue un **move**).

Cette définition garantit que chaque configuration de la séquence est suivie par une transition valide dans le système. Une **spécification**, dans ce contexte, correspond à un prédictat défini sur l'ensemble des exécutions, permettant de caractériser ou de contraindre les comportements souhaités du système.

Nous associons à chaque problème une spécification. Il s'agit d'un prédictat permettant de déterminer si une exécution d'un algorithme satisfait les contraintes du problème. Un système de transitions  $S = (C, \rightarrow)$  est **auto-stabilisant** pour la spécification  $P$  s'il existe un ensemble  $L \subseteq C$  de configurations **légitimes** vérifiant les propriétés suivantes :

1. **Correction** : toute exécution débutant par une configuration dans  $L$  satisfait  $P$ .

2. **Convergence** : toute exécution atteint une configuration dans  $L$ .

La **complexité temporelle** d'un algorithme peut être évaluée en nombre de **rounds** ou en nombre de **moves**. Un **round** représente une manière structurée d'analyser l'exécution d'un algorithme distribué. Ce concept a été introduit par [18] pour prendre explicitement en compte les nœuds activés. Un **round** est défini comme la plus petite sous-séquence d'une exécution telle que chaque nœud pouvant exécuter une instruction au début du round l'exécute au moins une fois ou devient inactif en raison de modifications externes (par exemple, des changements d'état de ses voisins) au cours le round.

Nous avons conçu des algorithmes auto-stabilisants pour construire un couplage dans un graphe, déterminer un ensemble indépendant et développer un opérateur permettant de transformer un algorithme d'un modèle à un autre.

**Description du travail publié:** Johanne Cohen, Jonas Lefèvre, Khaled Maâmra, George Manoussakis, and Laurence Pilard. "The first polynomial self-stabilizing 1-maximal matching algorithm for general graphs". *Theor. Comput. Sci.*, 782:54–78, 2019.

Nous avons conçu le premier algorithme auto-stabilisant polynomial permettant de trouver un 1-appariement maximal dans un graphe général. Le meilleur algorithme connu pour ce problème était celui proposé par Manne et al. [43], et nous montrons dans ce travail que sa complexité temporelle est sous-exponentielle sous le daemon distribué adversarial.

Notre nouvel algorithme est une adaptation de celui de Manne et al. [43] fonctionnant sous le même daemon, mais avec une complexité en  $O(nm)$  moves, où  $n$  est le nombre de nœuds et  $m$  le nombre d'arêtes. Il s'agit du premier algorithme auto-stabilisant résolvant ce problème avec une complexité polynomiale. De plus, notre algorithme ne requiert qu'une seule variable booléenne supplémentaire par rapport à l'algorithme précédent, garantissant ainsi une amélioration significative en termes d'efficacité et de simplicité.

**Description du travail publié:** Johanne Cohen, Laurence Pilard, and Jonas Sénizergues. "Self-stabilization and byzantine tolerance for maximal independent set." In *International Symposium on Stabilizing, Safety, and Security of Distributed Systems*, pages 479–483. Springer, 2021.

Nous avons analysé l'impact des pannes transitoires et des pannes byzantines sur la construction d'un ensemble indépendant maximal dans un réseau général. Nous avons adapté l'algorithme auto-stabilisant présenté par Turau [57] afin de calculer un tel ensemble de sommets. Notre algorithme est non seulement auto-stabilisant, tout en étant capable de fonctionner dans un contexte plus complexe où des pannes byzantines arbitraires peuvent survenir.

Les nœuds byzantins peuvent empêcher les nœuds voisins d'intégrer l'ensemble indépendant pendant une durée arbitrairement longue. Dans ce cadre, les nœuds byzantins peuvent empêcher leurs voisins d'intégrer l'ensemble indépendant pendant une durée arbitrairement longue. Pour limiter leur impact, nous avons introduit une variation du concept de **rayon de confinement**. À notre connaissance, nous proposons le premier algorithme tolérant à la fois les fautes transitoires et byzantines sous le daemon distribué équitable. Nous prouvons que cet algorithme converge en  $O(\log n)$  rounds avec une forte probabilité.

**Description du travail publié:** Johanne Cohen, Laurence Pilard, Mikaël Rabie, and Jonas Sénizergues. "Making self-stabilizing algorithms for any locally greedy problem". In *2nd Symposium on Algorithmic Foundations of Dynamic Networks SAND*, 2023.

Les algorithmes auto-stabilisants constituent une approche permettant de gérer la dynamique des réseaux, car ils s'adaptent automatiquement aux modifications structurelles du réseau, telles que l'ajout ou suppression de nœuds ou d'arêtes, à condition que ces modifications ne sont pas trop fréquentes.

Nous avons proposé une transformation automatique permettant de convertir des algorithmes distribués synchrones, qui résolvent des problèmes gloutons et réparables localement, en algorithmes auto-stabilisants adaptés aux réseaux anonymes.

Les problèmes réparables [4] généralisent les problèmes gloutons en permettant de transformer progressivement une solution partielle en une solution globale. Contrairement aux approches classiques où une solution partielle est simplement complétée, ici, chaque extension d'une solution partielle peut impliquer des modifications locales de la solution précédente, limitées à une distance donnée. Le terme **localement** signifie que l'extension d'une solution pour un nœud ne dépend que de son environnement immédiat, c'est-à-dire d'une zone de rayon constant autour de ce nœud.

Pour cela, nous proposons le premier algorithme auto-stabilisant explicite qui calcule un ensemble  $(k, k-1)$ -**ruling** (c'est-à-dire un ensemble indépendant maximal à distance  $k$ ). En appliquant plusieurs fois cette technique, nous obtenons une coloration à distance  $K$  du graphe. Grâce à cette coloration, nous pouvons enfin simuler des algorithmes du modèle Local, exécutés en un nombre constant de rounds, en utilisant les couleurs comme identifiants uniques.

Nos algorithmes fonctionnent sous le daemon de Gouda [22, 26], similaire au daemon probabiliste : si un événement doit se produire éventuellement, alors il surviendra.

**Description du travail publié:** Johanne Cohen, George Manoussakis, and Laurence Pilard. "From state to link-register model: A transformer for self-stabilizing distributed algorithms." In *2023 25th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)*, pages 114–121, 2023.

Dans le modèle à registres, il existe un délai entre le moment où une action est effectuée et celui où un nœud adjacent est informé de la modification résultante. Ce délai permet d'analyser l'asynchronisme induit par les communications dans les systèmes distribués.

L'atomicité lecture/écriture, qui est le modèle le plus restrictif, impose que les communications de nœud à nœud. De plus, dans ce cadre, le daemon distribué inéquitable peut retarder indéfiniment une communication tant qu'il existe encore d'autres moves possibles dans l'algorithme.

Nous avons proposé un transformateur permettant de convertir un algorithme auto-stabilisant du modèle état vers le modèle link-register.

Dans le pire des cas, un seul move d'un algorithme auto-stabilisant dans le modèle état peut générer  $\Delta$  rounds dans l'algorithme transformé dans le modèle link-register (où  $\Delta$  est le degré maximal du graphe).

Ce transformateur repose sur un autre transformateur, convertissant un algorithme du modèle état vers une version modifiée du modèle à registre, appelée **modèle à registre constraint**, où un nœud peut lire aussi dans ses propres registres. Cette transformation introduit un coût multiplicatif en  $O(\Delta)$ .

### Conception d'algorithmes appliqués à l'ordonnancement et aux télécommunications

**Description du travail publié:** Vinicius Freitas, Laércio Lima Pilla, Alexandre de L. Santana, Márcio Castro, and Johanne Cohen. "Packsteallb: A scalable distributed load balancer based on work stealing and workload discretization". *J. Parallel Distributed Comput.*, 150:34–45, 2021.

L'adaptabilité des applications pour des plate-formes pour des calculs à hautes performances est directement influencée par leur gestion des ressources informatiques. Ces applications sont souvent confrontées à des déséquilibres de charge, en raison de la nature dynamique de leurs calculs. Pour remédier à ce problème, les machines à haute performance mettent en œuvre périodiquement des stratégies d'équilibrage de charge.

Les algorithmes d'équilibrage de charge redistribuent la charge de travail en s'appuyant sur des heuristiques, permettant la complexité du problème. Cependant, les heuristiques d'ordonnancement doivent être rapides à exécuter afin de ne pas dégrader les performances de l'application lors de la distribution de la charge de travail dans des environnements étendus et distribués. Nous présentons une technique distribuée permettant de prendre des décisions d'ordonnancement à la fois efficaces et à faible coût pour les applications parallèles. Cette technique s'appuie sur des informations sur la charge de travail de l'application, associées à des algorithmes d'ordonnancement distribués. Les résultats expérimentaux ont montré que notre algorithme est capable d'améliorer les performances d'un benchmark de dynamique moléculaire.

**Description du travail publié:** Stephan Kunne, Lorenzo Maggi, Johanne Cohen, and Xu Xinneng. "Anytime Backtrack Unimodal Bandits and Applications to Cloud Computing", In *IFIP Networking Conference (Networking)*, number 4, pages 82–90, Paris, France, June 2020.

Nous sommes intéressés aux liens entre les algorithmes de non-regret [2, 37, 9] et des problèmes d'optimisation (trouver le minimum/maximum d'une fonction unimodale). Les fonctions unimodales sont omniprésentes dans les problèmes d'allocation de ressources. Elles apparaissent chaque fois que la valeur attendue de la fonction

$f$  à optimiser présente un "pic" unique en fonction de la variable de contrôle disponible  $x$ . Les propriétés et la définitions de ces fonctions ne sont souvent pas connue à l'avance, elles ne peuvent donc être apprises uniquement par échantillonnage. Dans ce but, nous exploitons les différentes techniques liés aux problèmes des bandits-manchots pour concevoir un algorithme d'apprentissage qui (1) converge vers le pic de la fonction et (2) qui ne nécessite pas de connaître le nombre d'échantillonnage à l'avance, c'est-à-dire qu'il est à tout moment. De plus, il s'adapte naturellement au scénario où une connaissance préalable sur la valeur du pic est disponible, même si cette connaissance est imprécise. Les algorithmes de l'état de l'art [65, 17, 16] n'étaient pas adapté à notre contexte (erreur dans les croyances, erreurs dans les échantillonnages).

## Étude du lien entre l'apprentissage automatique et l'aide à la décision

Ce travail s'inscrit dans le cadre de l'encadrement de deux thèses financées par deux collaborations CIFRE avec Thales Research & Technology France. Ces thèses ont été co-encadrées par Michèle Sébag (LISN) et Christophe Labreuche (Groupe Thales).

### Liens entre l'apprentissage et l'aide à la décision

L'aide à la décision multi-critères repose sur la conception de modèles permettant d'identifier ou de trouver des alternatives les plus pertinentes possibles pour un expert. Une **alternative** est définie comme un ensemble de valeurs associées à différents critères. Les problèmes résolus par ces modèles sont habituellement les suivants : choix (électionner la meilleure alternative), classement (ordonner les alternatives selon leur pertinence) et classification (attribuer chaque alternative à une classe de satisfaction).

Nous nous concentrons sur des modèles qui associent un score à une alternative donnée. Ce score permet de résoudre les trois problèmes. Nous travaillons sur l'**intégrale de Choquet (CI)** [15, 27]. Cette technique vise à attribuer à chaque alternative un score ou une utilité globale, reflétant son attractivité. Il est alors évident que ces scores peuvent servir de base pour classer les alternatives, établir des préférences par paires ou encore les répartir dans différentes catégories de préférence. Elle offre un bon compromis entre puissance de représentation (étant un agrégateur non linéaire capable de modéliser divers types d'interactions) et interprétabilité.

**Description du travail publié:** Roman Bresson, Johanne Cohen, Eyke Hüllermeier, Christophe Labreuche, and Michele Sebag. "Neural representation and learning of hierarchical 2-additive choquet integrals". In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence (IJCAI-20)*, pages 1984–1991, 2021.

Cependant, lorsque le nombre de critères atteint plusieurs dizaines, les modèles deviennent difficiles à concevoir et à interpréter. Les modèles hiérarchiques, et en particulier les **intégrales de Choquet hiérarchiques (HCI)**, permettent de surmonter cette limitation grâce à une approche de type "diviser pour régner", en agrégant progressivement les critères initiaux pour former des critères abstraits de plus haut niveau. Comme seul un petit nombre de critères est agrégé à chaque étape, les recommandations du modèle restent faciles à vérifier et à comprendre. De plus, étant donné que chaque CI individuel est monotone par rapport à ses entrées, nous avons la garantie que l'HCI hérite des mêmes propriétés.

Nous avons élaboré un outil, Neur-HCI, qui permet d'extraire automatiquement les paramètres d'un modèle HCI à partir de : (i) données disponibles, et (ii) structure d'agrégation hiérarchique définie sur les critères.

L'interprétabilité et la vérification du modèle sont garanties par le fait que Neur-HCI impose par construction toutes les contraintes du modèle HCI (par exemple, la monotonie et l'idempotence). Plus précisément, Neur-HCI traduit automatiquement la structure arborescente du modèle HCI en une architecture de réseau de neurones, dont les poids sont optimisés par rétropropagation à partir des données disponibles. Neur-HCI combine ainsi les avantages des modèles HCI, qui sont interprétables, et ceux des réseaux neuronaux, qui offrent une complexité d'apprentissage abordable.

**Description du travail publié:** Roman Bresson, Johanne Cohen, Eyke Hüllermeier, Christophe Labreuche, and Michèle Sebag. "On the identifiability of hierarchical decision models". In *Proceedings of the International Conference on Principles of Knowledge Representation and Reasoning (KR)*, volume 18, pages 151–162, 2021.

Nous avons pu démontrer l'identifiabilité de la classe des fonctions HCI. Autrement dit, sous des hypothèses modérées, une fonction HCI donnée possède une hiérarchie unique, un ensemble unique d'utilités marginales, ainsi qu'un ensemble unique de poids (paramètres de chaque agrégateur). Cette unicité est particulièrement intéressante, car elle renforce la confiance dans un modèle appris à partir de données statistiques.

## Liens entre l'explainabilité de l'apprentissage et l'aide à la décision.

Les réseaux neuronaux profonds (DNN) repoussent les limites de la performance dans de nombreux domaines, mais leur opacité les rend difficilement interprétables et limite la confiance des utilisateurs [49]. L'essor du domaine de l'IA explicable (XAI) est motivé par la nécessité de comprendre les DNN afin de pouvoir leur faire confiance, les déboguer ou les certifier.

**Description du travail publié:** Nicolas Atienza, Roman Bresson, Cyriaque Rousselot, Philippe Caillou, Johanne Cohen, Christophe Labreuche, and Michèle Sebag. "Cutting the black box: Conceptual interpretation of a deep neural net with multi-modal embeddings and multi-criteria decision aid". In *Thirty-Third International Joint Conference on Artificial Intelligence {IJCAI-24}*, pages 3669–3678. International Joint Conferences on Artificial Intelligence Organization, 2024.

Étant donné un DNN  $f$ , l'état de l'art en XAI vise à expliquer soit son résultat  $f(x)$  pour un échantillon particulier  $x$  (explication a posteriori), soit le modèle lui-même. Trois grandes approches émergent. Une première approche consiste à expliquer le modèle en identifiant les caractéristiques les plus influentes dans la décision. Celles-ci sont déterminées à partir du gradient de  $f(x)$  [48] ou des valeurs de Shapley [50]. Une seconde approche exploite des informations externes sur le domaine d'application, représentées sous forme de concepts et d'exemples illustratifs, et cherche à analyser leur impact dans l'espace latent du modèle [36]. Une troisième approche vise à caractériser les motifs récurrents associés à une classe spécifique, permettant ainsi d'identifier des régularités dans les données [10].

S'appuyant sur notre travail autour des intégrales de Choquet, nous proposons *Cut the Black Box (CB2)*, une approche qui fusionne la puissance des DNN et l'interprétabilité des modèles d'aide à la décision en vision par ordinateur. CB2 exploite des embeddings multimodaux pour projeter les représentations textuelles et visuelles dans un espace conceptuel partagé [47]. Ces embeddings, notamment ceux issus de *Contrastive Language-Image Pre-training (CLIP)* [47], sont devenus des références pour l'évaluation de modèles.

CB2 repose sur trois étapes. (1) Projection dans un espace conceptuel : Les embeddings CLIP permettent de cartographier les images en concepts définis par un dictionnaire spécifique au domaine; (2) Transfert de connaissances via distillation : Le modèle cible (enseignant), un DNN complexe, est utilisé pour entraîner un modèle MCDA étudiant, basé sur ces représentations conceptuelles; (3) Alignement des concepts intermédiaires : L'espace latent du modèle enseignant est mis en correspondance avec celui du modèle étudiant, rendant explicite la logique sous-jacente aux décisions du DNN. Ce processus ouvre la boîte noire en révélant pourquoi des exemples sont regroupés dans l'espace latent et partagent le même label. L'efficacité de CB2 est validée sur trois architectures de DNN – ResNet [30], VGG [53] et Vision Transformer (ViT) [21] – avec des expériences menées sur les benchmarks Cifar-10, Cifar-100, Tiny ImageNet et Fashion-MNIST.

## Propriétés structurelles sur les graphes et les jeux positionnels

**Description du travail publié:** David Auger, Johanne Cohen, and Antoine Lobstein. "Nonatomic Non-Cooperative Neighbourhood Balancing Games". *Fundamenta Informaticae*, 191(3-4):239–268, July 2024.

Les Jeux d'équilibrage de charge entre voisinage Non-Atomiques Non-Coopératifs sont des modèles de jeux non-coopératifs où des joueurs, considérés comme une masse continue, choisissent des ressources tout en supportant un coût qui dépend non seulement du nombre de joueurs ayant sélectionné la même ressource, mais aussi de ceux ayant choisi des ressources voisines.

Nous introduisons un jeu dans lequel les joueurs choisissent égoïstement une ressource et subissent un coût dépendant du nombre de joueurs sélectionnant des ressources voisines. Nous modélisons les influences entre les ressources par un graphe pondéré, orienté ou non. Ces jeux sont des généralisations de jeux bien connus tels que les jeux de Wardrop et les jeux de congestion. Nous étudions les conditions d'existence des équilibres et leur efficacité lorsqu'ils existent. Nous concluons par des études de jeux dont les influences entre les ressources peuvent être modélisées par des graphes simples.

Dans ces jeux, les influences entre les ressources sont représentées par un graphe pondéré, qui peut être dirigé ou non. Ce cadre général englobe des jeux bien connus tels que les jeux de congestion et le modèle de Wardrop. L'étude de ces jeux se concentre sur l'existence et l'efficacité des équilibres, c'est-à-dire des configurations où aucun joueur n'a intérêt à modifier sa stratégie unilatéralement. Les travaux récents ont également exploré des cas spécifiques où les influences entre les ressources peuvent être modélisées par des graphes simples.

## Absences de motifs pour les colorations acycliques

**Description du travail publié:** Quentin Chuet, Johanne Cohen, and François Pirot. "Acyclic colourings of graphs with obstructions". à paraître dans Discrete Mathematics en 2025

Étant donné un graphe  $G$ , une coloration de  $G$  est dite **acyclique** si elle est une coloration propre et si chaque cycle contient au moins trois couleurs. Le nombre chromatique acyclique  $\chi_a(G)$  est le plus petit entier  $k$  tel qu'il existe une  $k$ -coloration propre de  $G$  sans cycle bicolore.

Grünbaum [28] a introduit le concept de coloration acyclique, et A. V. Kostochka [38] a démontré que décider si un graphe  $G$  admet une  $k$ -coloration acyclique est un problème NP-complet. La plupart des recherches dans l'état de l'art se concentrent soit sur le calcul exact du nombre chromatique acyclique pour certaines familles de graphes [8], soit sur l'obtention de bornes supérieures pour sa valeur.

Fertin *et al.* [23] ont établi une borne inférieure simple sur le nombre chromatique acyclique :  $\chi_a(G) > \frac{d}{2} + 1$  lorsque  $G$  a un degré moyen  $d$ . Alon *et al.* [1] ont montré qu'il existe des graphes  $G$  de degré maximal  $\Delta(G)$  pour lesquels :  $\chi_a(G) = \Omega\left(\frac{\Delta(G)^{4/3}}{(\log \Delta(G))^{1/3}}\right)$  lorsque  $\Delta \rightarrow \infty$ . Ce résultat découle de l'analyse de graphes aléatoires dans le modèle d'Erdős-Rényi  $G(n, p)$ , où  $n$  sommets sont fixés et chaque arête existe avec une probabilité  $p = p(n)$ . En général, lorsque  $G$  a un degré maximal  $\Delta$ , il est connu que  $\chi_a(G) = O(\Delta^{4/3})$  lorsque  $\Delta \rightarrow \infty$ . Nous étudions l'effet de cette borne lorsqu'on impose en plus que  $G$  ne contienne pas un certain sous-graphe fixé  $F$  à  $t$  sommets.

Nous établissons que la borne est constante si  $F$  est un arbre subdivisé. Elle devient  $O(t^{8/3}\Delta^{2/3})$  si  $F$  est une forêt,  $O(\sqrt{t}\Delta)$  si  $F$  est biparti et 1-acyclique,  $2\Delta + o(\Delta)$  si  $F$  est un cycle de longueur paire d'au moins 6, et  $O(t^{1/4}\Delta^{5/4})$  si  $F = K_{3,t}$ . Ces résultats permettent d'affiner la compréhension du nombre chromatique acyclique en fonction des contraintes structurelles imposées au graphe  $G$ . Étant donné un graphe  $G$  et un entier  $k \geq 1$ , une  $k$ -coloration acyclique  $\phi$  de  $G$  est une  $k$ -coloration propre (une partition de  $V(G)$  en  $k$  ensembles indépendants appelés classes de couleurs de  $\phi$ ), telle que chaque paire de classes de couleurs induit une forêt (c'est-à-dire que tout cycle de longueur paire contient au moins trois couleurs).

## Modèles d'apprentissage et modèle de calculs

Plus récemment, mes recherches se sont orientées vers l'étude des modèles d'apprentissage eux-mêmes, au-delà de leurs applications. L'objectif est de mieux comprendre leurs fondements théoriques, leurs limites et leurs capacités de généralisation. Cette approche vise à établir des liens entre les modèles utilisés en pratique en apprentissage profond et les outils issus de la théorie des systèmes dynamiques (équations différentielles), la calculabilité et de la théorie de la complexité, afin de clarifier leur pouvoir expressif et leurs contraintes structurelles.

### Théorème d'approximation des réseaux de neurones étroits et profond

Les modèles d'apprentissage profond ont profondément transformé l'apprentissage automatique. L'un des résultats théoriques les plus connus est le *théorème d'approximation universelle* : pour toute fonction continue définie sur un domaine compact et pour toute précision  $\epsilon$ , il existe un réseau de neurones capable d'approximer cette fonction avec une erreur maximale d'au plus  $\epsilon$ . Autrement dit, la classe des fonctions représentées par les réseaux de neurones est dense dans l'ensemble des fonctions continues sur les compacts. Ce résultat s'obtient sous des hypothèses minimales (une seule couche cachée et une fonction d'activation non polynomiale suffisent [12, 32]), mais il implique de construire un réseau spécifique pour chaque fonction à approximer.

La littérature a largement exploré ce théorème et ses extensions, en étudiant les conditions sous lesquelles les réseaux atteignent l'approximation universelle en fonction de leur complexité structurelle. Celle-ci se mesure généralement par la profondeur (nombre de couches), la largeur (nombre de neurones par couche) et la taille totale (nombre de neurones). De nombreux travaux récents ont mis en évidence les compromis entre ces paramètres, montrant par exemple les bornes minimales de largeur nécessaires à l'approximation uniforme, ou encore l'efficacité accrue des réseaux profonds pour l'approximation de certaines classes de fonctions [29, 42, 64, 31].

Notre approche adopte une perspective différente : nous étudions l'*approximation uniforme universelle*, obtenue avec un réseau unique et fixe. Il est connu depuis les années 1990 que certains réseaux récurrents peuvent simuler des machines de Turing [51, 52], ce qui établit leur universalité au sens computationnel. Cependant, cela ne garantit pas leur universalité au sens de l'approximation uniforme.

Nous montrons qu'il est possible d'utiliser un seul et même réseau de neurones — de largeur bornée mais de profondeur arbitraire — pour approximer uniformément l'ensemble des fonctions calculables. Pour ce faire, nous relierons les questions d'approximation uniforme à celles de la calculabilité et de la théorie de la complexité, mettant en évidence la difficulté intrinsèque des problèmes associés. Nos résultats soulignent également l'écart encore

important entre les défis pratiques de l'apprentissage profond et leurs contreparties théoriques en calculabilité et en complexité.

**Description du travail publié:** Olivier Bournez, Johanne Cohen, Adrian Wurm. "A Universal Uniform Approximation Theorem for Neural Networks". In *50th International Symposium on Mathematical Foundations of Computer Science MFCS 2025*. LIPIcs, vol. 299, Article 29, pp. 29:1–29:20, 2025

### Modèle d'apprentissage continu : INode

Les techniques d'apprentissage automatique ont profondément renouvelé la modélisation des systèmes dynamiques, en exploitant données expérimentales et connaissances physiques. Deux approches dominent la littérature en apprentissage profond : la prévision de séries temporelles et l'identification directe de l'équation différentielle ordinaire (ODE) sous-jacente aux données [11, 58, 35, 44]. Une troisième, fondée sur la modélisation générative, sort du cadre de ce travail.

Notre motivation vient de la modélisation des accélérateurs de particules. Bien que des simulateurs précis existent, leur coût de calcul interdit une exploration systématique des nombreux réglages électromagnétiques nécessaires à la calibration. Des modèles de substitution ont été proposés [45, 5, 62, 55], mais nos expériences montrent qu'ils échouent à capturer le comportement hautement variable et parfois discontinu du faisceau, lié aux *paramètres de contrôle*.

Pour répondre à cette difficulté, nous proposons *INode*, une méthodologie inspirée de [11]. L'idée clé est de travailler non pas sur les données brutes, mais sur un encodage plus robuste, obtenu via des opérateurs appliqués aux entrées. En particulier, nous utilisons l'intégrale, numériquement plus stable que la dérivation [59], ce qui permet de lisser les discontinuités.

La méthode INode suit quatre étapes: (1) transformation des données en trajectoires intégrales continues; (2). apprentissage d'un modèle de régression prédictif reliant trajectoire intégrale et trajectoire originale; (3). résolution d'une ODE implicite où la trajectoire originale est remplacée par son approximation ; (4). reconstruction de la trajectoire originale à partir de la trajectoire intégrale approximée.

Les contributions principales sont triples. Contrairement aux Neural ODE classiques [11], INode permet de caractériser une famille paramétrée d'ODE, adaptée aux différents réglages de l'accélérateur. Nous montrons que, sous des hypothèses modérées, (i) l'ODE implicite admet une solution unique et (ii) la trajectoire originale peut être approximée avec une précision arbitraire dans la limite d'un grand nombre d'échantillons (propriété d'approximation universelle).

**Description du travail publié:** Johanne Cohen, Emmanuel Goutierre, Hayg Guler, Fatiros Kapotos, Sida-Bastien Li, Michèle Sebag, Bowen Zhu: "Modelling Dynamical Systems: Learning ODEs with No Internal ODE Resolution". In *18th International Conference on Reachability Problems, RP 2024*, pp. 221–237, 2024

## Références

- [1] Noga Alon, Colin McDiarmid, and Bruce Reed. Acyclic coloring of graphs. *Random Structures & Algorithms*, 2(3):277–288, 1991.
- [2] Peter Auer, Nicolò Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine Learning*, 47(2-3):235–256, 2002.
- [3] David Balduzzi, Marcus Frean, Lennox Leary, J. P. Lewis, Kurt Wan-Duo Ma, and Brian McWilliams. The shattered gradients problem: If resnets are the answer, then what is the question? In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 342–350. PMLR, 2017.
- [4] Alkida Balliu, Juho Hirvonen, Darya Melnyk, Dennis Olivetti, Joel Rybicki, and Jukka Suomela. Local mending. In *International Colloquium on Structural Information and Communication Complexity*, pages 1–20. Springer, 2022.
- [5] S. Biedron, L. Brouwer, D. L. Bruhwiler, N. M. Cook, A. L. Edelen, D. Filippetto, C. K. Huang, A. Huebl, T. Katsouleas, N. Kuklev, R. Lehe, S. Lund, C. Messe, W. Mori, C. K. Ng, D. Perez, P. Piot, J. Qiang, R. Roussel, D. Sagan, A. Sahai, A. Scheinker, M. Thévenet, F. Tsung, J. L. Vay, D. Winklehner, and H. Zhang. Snowmass21 accelerator modeling community white paper, 2022.

- [6] L. Blum, M. Shub, and S. Smale. On a theory of computation and complexity over the real numbers: NP-completeness, recursive functions and universal machines. *Bull. Amer. Math. Soc.*, 21(1):1–46, 1989.
- [7] Lenore Blum, Mike Shub, and Steve Smale. On a theory of computation and complexity over the real numbers; NP completeness, recursive functions and universal machines. *Bulletin of the American Mathematical Society*, 21(1):1–46, July 1989.
- [8] Oleg V Borodin. On acyclic colorings of planar graphs. *Discrete Mathematics*, 25(3):211–236, 1979.
- [9] Olivier Cappé, Aurélien Garivier, Odalric-Ambrym Maillard, Rémi Munos, Gilles Stoltz, et al. Kullback–leibler upper confidence bounds for optimal sequential allocation. *The Annals of Statistics*, 41(3):1516–1541, 2013.
- [10] Chaofan Chen, Oscar Li, Daniel Tao, Alina Barnett, Cynthia Rudin, and Jonathan K Su. This looks like that: deep learning for interpretable image recognition. *Advances in neural information processing systems*, 32, 2019.
- [11] Tian Qi Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. In *Advances in Neural Information Processing Systems*, pages 6571–6583, 2018.
- [12] George Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of control, signals and systems*, 2(4):303–314, 1989.
- [13] David Maxwell Chickering. Learning bayesian networks is np-complete. *Learning from data: Artificial intelligence and statistics V*, pages 121–130, 1996.
- [14] Kyunghyun Cho, Bart van Merriënboer, Çağlar Gülcehre, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *CoRR*, abs/1406.1078, 2014.
- [15] Gustave Choquet. Theory of capacities. In *Annales de l'institut Fourier*, volume 5, pages 131–295, 1954.
- [16] Richard Combes and Alexandre Proutière. Unimodal bandits: Regret lower bounds and optimal algorithms. In *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014*, volume 32 of *JMLR Workshop and Conference Proceedings*, pages 521–529. JMLR.org, 2014.
- [17] Richard Combes and Alexandre Proutière. Unimodal bandits without smoothness. *CoRR*, abs/1406.7447, 2014.
- [18] Alain Cournier, Stéphane Devismes, and Vincent Villain. Snap-stabilizing pif and useless computations. In *12th International Conference on Parallel and Distributed Systems-(ICPADS'06)*, volume 1, pages 8–pp. IEEE, 2006.
- [19] Shlomi Dolev. Self stabilization. *Journal of Aerospace Computing, Information, and Communication*, 1(6):253–255, 2004.
- [20] Shuyu Dong, Michèle Sebag, Kento Uemura, Akito Fujii, Shuang Chang, Yusuke Koyanagi, and Koji Maruhashi. Dcdilp: a distributed learning method for large-scale causal structure learning. *arXiv preprint arXiv:2406.10481*, 2024.
- [21] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words. *arXiv preprint arXiv:2010.11929*, 7, 2020.
- [22] Swan Dubois and Sébastien Tixeuil. A taxonomy of daemons in self-stabilization. *arXiv preprint arXiv:1110.0334*, 2011.
- [23] Guillaume Fertin, Emmanuel Godard, and André Raspaud. Acyclic and k-distance coloring of the grid. *Information Processing Letters*, 87(1):51–58, 2003.
- [24] Tian Gao, Kshitij Fadnis, and Murray Campbell. Local-to-global bayesian network structure learning. In *International Conference on Machine Learning*, pages 1193–1202. PMLR, 2017.

- [25] Aidan N. Gomez, Mengye Ren, Raquel Urtasun, and Roger B. Grosse. The reversible residual network: Backpropagation without storing activations. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, volume 30, pages 2214–2224, 2017.
- [26] Mohamed G Gouda. The theory of weak stabilization. In *Self-Stabilizing Systems: 5th International Workshop, WSS 2001 Lisbon, Portugal, October 1–2, 2001 Proceedings* 5, pages 114–123. Springer, 2001.
- [27] Michel Grabisch. L'utilisation de l'intégrale de choquet en aide multicritère à la décision. *Newsletter of the European Working Group "Multicriteria Aid for Decisions*, 3(14):5–10, 2006.
- [28] Branko Grünbaum. Acyclic colorings of planar graphs. *Israel journal of mathematics*, 14(4):390–408, 1973.
- [29] Boris Hanin and Mark Sellke. Approximating continuous functions by ReLU nets of minimal width. *arXiv preprint arXiv:1710.11278*, 2017.
- [30] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
- [31] Ruiyang Hong and Anastasis Kratsios. Bridging the gap between approximation and learning via optimal approximation by ReLU MLPS of maximal regularity. *arXiv preprint arXiv:2409.12335*, 2024.
- [32] Kurt Hornik. Approximation capabilities of multilayer feedforward networks. *Neural networks*, 4(2):251–257, 1991.
- [33] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.
- [34] Patrick Kidger. On neural differential equations. *CoRR*, abs/2202.02435, 2022.
- [35] Patrick Kidger, James Morrill, James Foster, and Terry Lyons. Neural controlled differential equations for irregular time series. In *Advances in Neural Information Processing Systems*, volume 33, pages 6696–6707. Curran Associates, Inc.
- [36] Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, et al. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *International conference on machine learning*, pages 2668–2677. PMLR, 2018.
- [37] Robert Kleinberg. Anytime algorithms for multi-armed bandit problems. In *Proceedings of the seventeenth annual ACM-SIAM symposium on Discrete algorithm*, pages 928–936. Society for Industrial and Applied Mathematics, 2006.
- [38] Alexandr V Kostochka. Upper bounds of chromatic functions of graphs. In *Doct. Thesis*. Novosibirsk, 1978.
- [39] Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. Fractalnet: Ultra-deep neural networks without residuals. *ICLR*, 2016.
- [40] Po-Ling Loh and Peter Bühlmann. High-dimensional learning of linear causal networks via inverse covariance estimation. *The Journal of Machine Learning Research*, 15(1):3065–3105, 2014.
- [41] Yiping Lu, Aoxiao Zhong, Quanzheng Li, and Bin Dong. Beyond finite layer neural networks: Bridging deep architectures and numerical differential equations. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Workshop Track Proceedings*, pages 3276–3285. PMLR, OpenReview.net, 2018.
- [42] Zhou Lu, Hongming Pu, Feicheng Wang, Zhiqiang Hu, and Liwei Wang. The expressive power of neural networks: A view from the width. *Advances in neural information processing systems*, 30, 2017.
- [43] Fredrik Manne, Morten Mjelde, Laurence Pilard, and Sébastien Tixeuil. A self-stabilizing 23-approximation algorithm for the maximum matching problem. *Theoretical Computer Science*, 412(40):5515–5526, 2011.

- [44] Stefano Massaroli, Michael Poli, Jinkyoo Park, Atsushi Yamashita, and Hajime Asama. Dissecting neural odes. *Advances in Neural Information Processing Systems*, 33:3952–3963, 2020.
- [45] John A Miller, Mohammed Aldosari, Farah Saeed, Nasid Habib Barna, Subas Rana, I Budak Arpinar, and Ninghao Liu. A survey of deep learning and foundation models for time series forecasting. *arXiv preprint arXiv:2401.13912*, 2024.
- [46] Leland Gerson Neuberg. Causality: models, reasoning, and inference, by judea pearl, cambridge university press, 2000. *Econometric Theory*, 19(4):675–685, 2003.
- [47] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.
- [48] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144, 2016.
- [49] Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature machine intelligence*, 1(5):206–215, 2019.
- [50] M Scott, Lee Su-In, et al. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30:4765–4774, 2017.
- [51] Hava T. Siegelmann and Eduardo D. Sontag. Turing computability with neural nets. *Applied Mathematics Letters*, 4(6):77–80, 1991.
- [52] Hava T. Siegelmann and Eduardo D. Sontag. Analog computation via neural networks. *Theoretical Computer Science*, 131(2):331–360, sep 1994.
- [53] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.
- [54] Peter Spirtes, Clark Glymour, and Richard Scheines. *Causation, prediction, and search*. MIT press, 2001.
- [55] Kunxiang Sun, Xiaolong Chen, Xiaoying Zhao, Xin Qi, Zhijun Wang, and Yuan He. Surrogate model of particle accelerators using encoder-decoder neural networks with physical regularization. *International Journal of Modern Physics A*, 38:2350145–2928, 2023.
- [56] Ioannis Tsamardinos, Constantin F Aliferis, Alexander R Statnikov, and Er Statnikov. Algorithms for large scale markov blanket discovery. In *FLAIRS*, volume 2, pages 376–81, 2003.
- [57] Volker Turau. Linear self-stabilizing algorithms for the independent and dominating set problems using an unfair distributed scheduler. *Information Processing Letters*, 103(3):88–93, 2007.
- [58] Belinda Tzen and Maxim Raginsky. Neural stochastic differential equations: Deep latent gaussian models in the diffusion limit, 2019.
- [59] Bernd Ullmann. *Analog and hybrid computer programming*. Walter de Gruyter GmbH & Co KG, 2023.
- [60] Klaus Weihrauch. *Computable Analysis - An Introduction*. Texts in Theoretical Computer Science. An EATCS Series. Springer, 2000.
- [61] E Weinan. A proposal on machine learning via dynamical systems. *Communications in Mathematics and Statistics*, 1(5):1–11, 2017.
- [62] Liling Xiao, Lixin Ge, Zenghai Li, and Cho-Kuen Ng. Advances in multiphysics modeling for parallel finite-element code suite ace3p. *IEEE Journal on Multiscale and Multiphysics Computational Techniques*, 4:298–306, 2019.
- [63] Hanshu Yan, Jiawei Du, Vincent Y. F. Tan, and Jiashi Feng. On robustness of neural ordinary differential equations. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020.

- [64] Dmitry Yarotsky. Error bounds for approximations with deep ReLU networks. *Neural networks*, 94:103–114, 2017.
- [65] Jia Yuan Yu and Shie Mannor. Unimodal bandits. In Lise Getoor and Tobias Scheffer, editors, *Proceedings of the 28th International Conference on Machine Learning, ICML 2011, Bellevue, Washington, USA, June 28 - July 2, 2011*, pages 41–48. Omnipress, 2011. Available at [http://www.icml-2011.org/papers/50\\_icmlpaper.pdf](http://www.icml-2011.org/papers/50_icmlpaper.pdf).
- [66] Xingcheng Zhang, Zhizhong Li, Chen Change Loy, and Dahua Lin. Polynet: A pursuit of structural diversity in very deep networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 3900–3908. IEEE Computer Society, 2017.