

Axel Kerinec

2 bis rue Fénelon
44000 Nantes, France
T +33 6 23 02 39 62
E axel.kerinec@yahoo.com
W <https://axelkrnc.github.io/>

Formation Universitaire

- Master Informatique, École Normale Supérieure de Lyon. 2017-2018**
Automates, coinduction et algèbre relationnelle - logique monadique du second ordre - automate, expressivité et décidabilité - complexité computationnelle implicite - modèles de concurrence, catégories et jeux - réseaux complexes - méthodes de bornes inférieures - décompositions de graphes : de tree-Width aux graphes parfaits.
- Master Informatique, École Normale Supérieure de Lyon. 2016-2017**
Algorithmes distribués et parallèles - théorie de l'information - optimisation et approximation - évaluation des performances de réseaux - sémantique et vérification - preuves et programmes - bases de données et data mining - complexité computationnelle - machine learning.
- Licence Informatique, École Normale Supérieure de Lyon. 2015-2016**
Théorie des langages de programmation - machines de Turing et automates - logique mathématique - probabilité - architecture, systèmes et réseaux - design, complexité, implémentation algorithmique.
- Classes préparatoires MP/MPSI, Centre International de Valbonne. 2013-2015**
Option informatique.
- Baccalauréat Scientifique, Externat Saint Joseph - La Cordeille. 2010-2013**
Filière science de l'ingénieur et option mathématiques.

Expérience en Enseignement

J'ai réalisé une mission doctorale d'enseignement puis des vacances. Mes missions consistaient principalement à donner des TPs/TDs et à corriger des examens. En 2021-2022, j'ai encadré des des projets SAE (Situation d'Apprentissage et d'Évaluation) en BUT : des projets pratiques sur plusieurs semaines en petit groupe où un problème réaliste est donné aux élèves qui doivent analyser les besoins, prévoir une réponse adéquate et la mettre en œuvre. Cette même année j'ai également été membre de jurys pour des soutenances de stages. En 2024-2025, j'ai été responsable d'un chapitre du cours de logique pour l'informatique en 2ème année de licence : j'ai conçu le contenu du cours, les slides de présentation associées ainsi que les exercices destinés aux TDs.

- Université de Nantes département informatique. 2024-2025**
- Logique pour l'informatique, 2ème année de licence semestre 1 : 22h TD et 3h CM.
ma présentation pour le cours : https://axelkrnc.github.io/includes/cours_logique.pdf.
- Université de Nantes département informatique. 2023-2024**
- Mathématiques pour les informaticiens, 1ère année de licence semestre 1 : 24h TD et 16h TP.
- BUT Villetaneuse département d'informatique. 2021-2022**
- Encadrement de projets SAE, 1ère année semestre 2.
 - Jury pour des soutenances de stage, 2ème année semestre 2.
 - Introduction aux systèmes d'exploitation, 1ère année semestre 1 : 28,5h TP.
 - Bases de données avancées, 2ème année semestre 1 : 31h TP.
- DUT Villetaneuse département d'informatique. 2020-2021**
- Architecture et programmation, 1ère année semestre 2 : 12h TP.
 - Introduction à l'algorithmie et à la programmation (Python), 1ère année semestre 1 : 32h TP.
 - Conception de documents et d'interfaces numérique, 1ère année semestre 1 : 18h TP.
- DUT Villetaneuse département d'informatique. 2019-2020**
- Architecture et programmation, 1ère année semestre 2 : 21h TD.
 - Services réseaux, 2ème année semestre 1 : 23h TD.
 - Structures de données et algorithmes fondamentaux (Python), 1ère année semestre 1 : 16h TP.
- Auteur H&K publications. 05/2018**
Rédaction d'une version corrigée de sujets de concours d'entrée aux grandes écoles.

Expérience en Recherche

Post-Doctorat

Capture sémantique des coûts en temps et espace de l'évaluation pour des langages de programmation d'ordres supérieurs, 09/2023-actuel

INRIA et Laboratoire des Sciences du Numérique de Nantes,

Un projet dirigé par Guilhem Jaber.

La sémantique de jeu opérationnelle (OGS), introduite dans [Laird, 2007], interprète les termes du λ -calcul comme des états évoluant grâce à des systèmes de transitions labellisés, représentant les interactions avec l'environnement.

Contrairement à la sémantique de jeu dénotationnelle standard, la composition n'est pas une notion primitive. Cependant la plupart des modèles OGS considèrent une notion de composition fermée entre deux états, où seules les actions sans aucune interaction avec le monde extérieur peuvent être effectuées. Mais cela n'est pas suffisant pour construire un modèle compositionnel et le rendre adéquat : une notion de composition ouverte est nécessaire. Dans ce cas, la fusion d'une paire d'états ne donne plus un terme fermé, mais un autre état qui à la fois effectue des actions de synchronisation internes et est capable d'interagir avec l'environnement au moyen d'actions externes. Notre travail vise à définir une telle composition et à l'utiliser pour introduire des structures catégoriques pertinentes pour modéliser le λ -calcul.

Doctorat

Une histoire de λ -calcul et d'approximation, 09/2019-06/2023

Université Sorbonne Paris Nord.

Directeur : Giulio Manzonetto (Université Sorbonne Paris Nord).

Rapporteurs : Silvia Ghilezan (Université de Novi Sad),
Tom Hirschowitz (Université Savoie Mont Blanc).

Jury : Pierre Clairambault (Aix-Marseille Université),
Delia Kesner (Université Paris Cité),
Stefano Guerrini (Université Sorbonne Paris Nord),
Marie Kerjean (Université Sorbonne Paris Nord).

Manuscrit : <https://theses.hal.science/tel-04624826>.

Slides : <https://axelkrnc.github.io/includes/PhD.pdf>.

Sujet : Les arbres de Böhm [Barendregt, 1977] sont historiquement la première notion d'approximation pour le λ -calcul. Ils jouissent d'un lien étroit avec une autre notion d'approximation fondamentale : le développement de Taylor [Ehrhard and Regnier, 2003]. Ces notions sont initialement introduites en Appel-par-Nom, et la théorie de l'approximation en Appel-par-Valeur [Plotkin, 1975] est beaucoup moins développée, bien que cette stratégie de réduction soit plus proche des langages de programmations actuels.

Dans ce contexte, nous présentons la première notion d'arbre de Böhm. Et nous montrons une connexion entre nos arbres de Böhm et le développement de Taylor similaire à celle en Appel-par-Nom, prouvant ainsi la pertinence de notre définition. Nos arbres de Böhm permettent, entre autres, de caractériser l'équivalence observationnelle et la solvabilité, deux notions qui bien qu'essentielles sont peu comprises actuellement dans le cadre Appel-par-Valeur.

Dans la seconde partie de ce travail, nous nous intéressons à l'Appel-par-Nom. Une piste de recherche récente est d'étudier des modèles dans des catégories d'ordre supérieurs, ceux-ci sont prometteurs car ils présentent plus d'informations que les modèles classiques.

Dans la lignée de [Olimpieri, 2020, Olimpieri, 2021], nous introduisons des modèles qui peuvent être vus comme une catégorification des modèles de graphe relationnels traditionnels et pareillement ils peuvent être présentés comme des systèmes de types intersection. Nos modèles bénéficient d'un théorème d'approximation : l'interprétation d'un λ -terme correspond à l'interprétation de son arbre de Böhm. Ce résultat fondamental est démontré par simple induction. Contrairement aux modèles relationnels, nos modèles sont également preuve-dépendants, dans le sens où l'interprétation d'un λ -terme dans nos modèles ne contient pas seulement ses typages, mais également l'ensemble des dérivations de types. Grâce aux informations supplémentaires portées par les dérivations de types nous obtenons la caractérisation de la théorie induite comme un simple corollaire du théorème d'approximation : deux λ -termes ont des interprétations isomorphes exactement lorsque leurs arbres de Böhm coïncident.

Stages

Conservativité dans le λ -calcul algébrique, 03/2019-08/2019

Institut de Mathématiques de Marseille,

Supervisé par Lionel Vaux Auclair.

Le λ -calcul algébrique a été introduit comme un cadre général pour étudier la théorie de la réécriture des λ -termes en présence de superpositions pondérées [Vaux, 2007, Vaux, 2009]. Un résultat depuis longtemps avancé est celui de la conservativité : deux λ -termes classiques sont équivalents dans le cadre algébrique si et seulement si ils sont β -égaux. Cependant les preuves précédentes de ce résultat sont fausses, durant ce stage nous étudions pourquoi et développons une nouvelle preuve en utilisant une technique originale.

λ -calcul probabiliste et méthodes de Monte Carlo, 10/2018-03/2019

Université de Bologne,

Supervisé par Ugo Dal Lago.

Nous nous intéressons au λ -calcul probabiliste. En particulier nous considérons le langage de programmation fonctionnel typé "Programming Computable Functions" qui peut être vu comme une version étendue du λ -calcul [Plotkin, 1977] et que nous modifions pour devenir probabiliste. Dans ce cadre lorsqu'on observe les résultats possibles de la réduction d'un λ -terme on obtient une distribution et non un seul λ -terme. Ces distributions ne peuvent pas être calculées exactement d'où la nécessité d'utiliser des algorithmes d'inférence pour les approximer. Nous nous sommes penchés sur l'algorithme de Monte Carlo séquentiel [Del Moral, 1997, Del Moral et al., 2006] ainsi que sur un cas spécifique de chaînes de Markov Monte Carlo : Metropolis-Hastings [Keith Hastings, 1970, Metropolis et al., 1953] et sur la méthode Particle Markov chain Monte Carlo [Christophe Andrieu, 2010]. Les deux premières méthodes sont classiques et ont déjà été utilisées pour l'étude de langages de programmation probabilistes, la dernière méthode est une combinaison des deux précédentes et est plus originale.

Approximation de λ -termes en Appel-par-Valeur, 02/2018-06/2018

IRIF, Université Paris Diderot,

Supervisé par Michele Pagani et Giulio Manzonetto.

Le λ -calcul en Appel-par-Valeur [Plotkin, 1975] a une théorie bien moins développée que celle du λ -calcul en Appel-par-Nom. Le développement de Taylor est une méthode d'approximation classique connue dans ces deux cadres [Ehrhard and Regnier, 2003] mais beaucoup plus étudiée dans le cadre de l'Appel-par-Nom, où il a notamment un lien fort avec les arbres de Böhm [Barendregt, 1977], qui n'existent pas dans le cadre en Appel-par-Valeur. Au cours de ce stage nous nous employons à combler ce retard en Appel-par-Valeur.

Utilité de l'apprentissage multi-tâches pour les réseaux de neurones, 05/2017-09/2017

Université de Copenhague,

Supervisé par Anders Søgaard.

Ce stage questionnait le gain apporté par l'apprentissage multi-tâches, dans le cadre des réseaux neuronaux profonds, comparé à l'apprentissage tâche-simple classique. Des études ont déjà étudié empiriquement quand l'apprentissage multi-tâches impliquait un gain ou non [Bingel and Søgaard, 2017, Alonso and Plank, 2016]. Cependant nous sommes les premiers à étudier cette question dans le cadre de la classification de textes, une tâche pour laquelle on emploie pourtant très souvent du machine learning.

Détection de motifs spatio-temporels dans des signaux multi-variés, 06/2016-08/2016

LabSTICC, École Telecom Bretagne,

Supervisé par Vincent Gripon.

La détection de motifs récurrents dans des images ou des vidéos est un enjeu majeur. En considérant les images comme des graphes nous pouvons calculer leurs pseudo-transformées de Fourier. La question de ce stage était de savoir si l'utilisation de cette transformée sur les données améliorerait ou non la détection avec la méthode d'apprentissage des machines à vecteurs de support.

Publications

À noter : Certaines de mes publications utilisent mon ancien prénom : Emma.

Abréviations :

HOR International Workshop on Higher-Order Rewriting

POPL Symposium on Principles of Programming Languages (ranks as A* by CORE)

FSCD International Conference on Formal Structures for Computation and Deduction

LMCS Logical Methods in Computer Science

Why Are Proofs Relevant in Proof-Relevant Models?

POPL 2023

[Kerinec et al., 2023],

Auteurs : A. Kerinec, G. Manzonetto et F. Olimpieri,

Lien : <https://dl.acm.org/doi/pdf/10.1145/3571201>.

Sujet : Nous étudions une classe de modèles au sein d'une sémantique bicatégorique qui généralise la sémantique relationnelle. Comme les modèles relationnels ils peuvent être présentés sous forme de systèmes de types et satisfont un théorème d'approximation qui peut être démontré par simple induction. Contrairement aux modèles relationnels, nos modèles sont aussi preuves-dépendants : l'interprétation d'un λ -terme ne contient pas seulement ses typages, mais les dérivations de types entières. Cette information additionnelle permet de caractériser simplement la λ -théorie induite, deux λ -termes ont des interprétations isomorphes exactement quand leurs arbres de Böhm coïncident.

Call-By-Value, Again! [Kerinec et al., 2021],

FSCD 2021

Auteurs : A. Kerinec, G. Manzonetto et S. Ronchi Della Rocca,

Lien : <https://drops.dagstuhl.de/storage/00lipics/lipics-vol195-fscd2021/LIPIcs.FSCD.2021.7/LIPIcs.FSCD.2021.7.pdf>.

Sujet : A ce jour aucun modèle complètement adéquat n'existe pour le λ -calcul en Appel-par-Valeur. Ici nous étudions un modèle relationnel présenté comme un système de types intersection, où l'intersection est non-idempotente sauf pour un élément spécifique. Ce modèle est adéquat, égalise beaucoup de λ -termes qui sont équivalents dans la théorie observationnelle maximale et satisfait un théorème d'approximation par rapport aux approximants des arbres de Böhm Appel-par-Valeur. Par ce prisme nous nous intéressons aux propriétés de la solvabilité et valubabilité (-potentielle).

Revisiting Call-by-value Böhm trees in light of their Taylor expansion

LMCS 2020

[Kerinec et al., 2020],

Auteurs : A. Kerinec, G. Manzonetto et M. Pagani,

Lien : <https://arxiv.org/abs/1809.02659>.

Sujet : Nous introduisons la première notion satisfaisante d'arbre de Böhm dans le cadre de l'Appel-par-Valeur. Notamment tous les λ -termes avec le même arbre de Böhm sont équivalents observationnellement. Nous comparons notre approche avec la théorie de l'approximation d'Ehrhard basée sur le développement de Taylor des λ -termes et trouvons un lien similaire entre ces deux notions que dans le cadre Appel-par-Nom original.

Papiers présentés à des workshops

The algebraic λ -calculus is a conservative extension of the ordinary

HOR 2023

λ -calculus [Kerinec and Vaux Auclair, 2023],

Auteurs : A. Kerinec et L. Vaux Auclair,

Lien : <https://hal.science/hal-04759250>.

Sujet : Le λ -calcul algébrique est une extension du λ -calcul classique avec des combinaisons linéaires de termes. Nous prouvons que deux λ -termes ordinaires sont équivalents dans le λ -calcul algébrique si et seulement si ils sont β -égaux. Ce résultat a originalement été annoncé durant les années 2000 mais les preuves précédentes se sont avérées fausses pour des raisons que nous détaillons.

When does deep multi-task learning work for loosely related document classification tasks? [Kerinec et al., 2018],

BlackboxNLP 2018

Auteurs : A. Kerinec, C. Braud et A. Søgaard,

Lien : <https://hal.science/hal-02374086v1>.

Sujet : Le but de ce travail est une meilleure compréhension de quand l'apprentissage multi-tâches par des réseaux neuronaux profonds qui partagent des paramètres est plus efficace que l'apprentissage tâche-simple. Nous sommes les premiers à effectuer ce travail pour de la classification de textes.

Exposés

Conférences

FSCD 2021, 20/07/2021
Call-By-Value, Again!
Slides : <https://axelkrnc.github.io/includes/fscd.pdf>
Vidéo : <https://www.youtube.com/watch?v=K06e-xlCH-c&t=356s>

Séminaires d'équipe

Équipe Logique et Interactions (I2M, Université Aix-Marseille), 08/09/2022
Modèles Bicatégoriques.

Slides : <https://axelkrnc.github.io/includes/i2m.pdf>

Equipe LoVe (LIPN, Université Sorbonne Paris Nord), 30/09/2021
Call-By-Value, Again!

Slides : <https://axelkrnc.github.io/includes/lipn.pdf>

Autres

LHC (Logic, Homotopy, Categories), 05/06/2024
Towards Categorical Structures for Operational Game Semantics.

Slides : <https://axelkrnc.github.io/includes/LHC.pdf>

Rencontres "CHoCoLa" (Curry-Howard : Calcul et Logique), 11/05/2023
Why Are Proofs Relevant in Proof-Relevant Models?.

Slides : <https://axelkrnc.github.io/includes/chocola.pdf>

Responsabilités

Aide à l'organisation de la conférence Microservices 2022, 10 au 12 mai 2022
[https://www.conf-micro.services/2022/.](https://www.conf-micro.services/2022/)

Compétences Informatiques

Programmation : OCaml, C, C++, Python

Web : HTML5, CSS3, SQL

Autres : Git, Unity, Latex

Références

- [Alonso and Plank, 2016] Alonso, H. M. and Plank, B. (2016). Multitask learning for semantic sequence prediction under varying data conditions. *CoRR*, abs/1612.02251.
- [Barendregt, 1977] Barendregt, H. P. (1977). The type free lambda calculus. In Barwise, J., editor, *Handbook of Mathematical Logic*, volume 90 of *Studies in Logic and the Foundations of Mathematics*, pages 1091–1132. North-Holland, Amsterdam.
- [Bingel and Sjøgaard, 2017] Bingel, J. and Sjøgaard, A. (2017). Identifying beneficial task relations for multi-task learning in deep neural networks.
- [Christophe Andrieu, 2010] Christophe Andrieu, A. D. (2010). Particle markov chain monte carlo methods. In *journal of the royal statistical society*.
- [Del Moral, 1997] Del Moral, P. (1997). Nonlinear filtering: Interacting particle resolution. *Comptes Rendus de l'Académie des Sciences - Series I - Mathematics*.
- [Del Moral et al., 2006] Del Moral, P., Doucet, A., and Jasra, A. (2006). Sequential monte carlo samplers. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(3):411–436.
- [Ehrhard and Regnier, 2003] Ehrhard, T. and Regnier, L. (2003). The differential lambda-calculus. *Theor. Comput. Sci.*, 309(1-3):1–41.
- [Keith Hastings, 1970] Keith Hastings, W. (1970). Monte carlo sampling methods using markov chains and their application. *Biometrika*.
- [Kerinec et al., 2023] Kerinec, A., Manzonetto, G., and Olimpieri, F. (2023). Why are proofs relevant in proof-relevant models? *Proc. ACM Program. Lang.*, 7(POPL):218–248.
- [Kerinec et al., 2020] Kerinec, A., Manzonetto, G., and Pagani, M. (2020). Revisiting call-by-value Böhm trees in light of their Taylor expansion. *Log. Methods Comput. Sci.*, 16(3).
- [Kerinec et al., 2021] Kerinec, A., Manzonetto, G., and Ronchi Della Rocca, S. (2021). Call-by-value, again! In Kobayashi, N., editor, *6th International Conference on Formal Structures for Computation and Deduction, FSCD 2021, July 17-24, 2021, Buenos Aires, Argentina (Virtual Conference)*, volume 195 of *LIPICs*, pages 7:1–7:18. Schloss Dagstuhl - Leibniz-Zentrum für Informatik.
- [Kerinec and Vaux Auclair, 2023] Kerinec, A. and Vaux Auclair, L. (2023). The algebraic λ -calculus is a conservative extension of the ordinary λ -calculus. *arXiv e-prints*, page arXiv:2305.01067.
- [Kerinec et al., 2018] Kerinec, E., Sjøgaard, A., and Braud, C. (2018). When does deep multi-task learning work for loosely related document classification tasks? In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 1 – 8, Brussels, Belgium. Association for Computational Linguistics.
- [Laird, 2007] Laird, J. (2007). A fully abstract trace semantics for general references. In Arge, L., Cachin, C., Jurdzinski, T., and Tarlecki, A., editors, *Automata, Languages and Programming, 34th International Colloquium, ICALP 2007, Wroclaw, Poland, July 9-13, 2007, Proceedings*, volume 4596 of *Lecture Notes in Computer Science*, pages 667–679. Springer.
- [Metropolis et al., 1953] Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953). Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*.
- [Olimpieri, 2020] Olimpieri, F. (2020). Intersection type distributors. *CoRR*, abs/2002.01287.
- [Olimpieri, 2021] Olimpieri, F. (2021). Intersection type distributors. In *36th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2021, Rome, Italy, June 29 - July 2, 2021*, pages 1–15. IEEE.
- [Plotkin, 1977] Plotkin, G. (1977). Lcf considered as a programming language. *Theoretical Computer Science*.

- [Plotkin, 1975] Plotkin, G. D. (1975). Call-by-name, call-by-value and the lambda-calculus. *Theor. Comput. Sci.*, 1(2):125–159.
- [Vaux, 2007] Vaux, L. (2007). On linear combinations of λ -terms. In Baader, F., editor, *Term Rewriting and Applications*, pages 374–388, Berlin, Heidelberg. Springer Berlin Heidelberg.
- [Vaux, 2009] Vaux, L. (2009). The algebraic lambda-calculus. *Mathematical Structures in Computer Science*, page accepted for publication. 29 pages.