- #1 Flexible ultra low power architecture supporting different artificial intelligence algorithms in the Internet of Things context

Titre du projet de recherche

Architectures de calcul flexibles et à très faible consommation d'énergie pour l'implémentation d'algorithmes variés d'intelligence artificielle dans le contexte de l'internet des objets

Niveau

Doctorat

Dates du projet

Du 1er octobre 2023 au 30 septembre 2026

Financement

Bourse du Ministère de l'enseignement supérieur, de la recherche et de l'innovation (MESRI) (environ 2045 € bruts/mois) pour trois ans.

Encadrants

Sébastien Pillement (IETR), Andrea Pinna (LIP6) et Pierre Langlois (Polytechnique Montréal)

Mise en contexte

Ce projet de thèse se situe dans le cadre de l'action Adaptive architectures for embedded artificial INtelliGence (AdaptING) du programme PEPR-IA, financée à hauteur de 5.6 M€ pour les années 2023-2026. Six équipes situées à Nantes, Rennes, Brest, Grenoble, Lyon et Paris font partie de l'action. L'action AdaptING se situe au croisement entre les domaines de l'informatique "architectures des machines" et apprentissage automatique. Elle vise à faire progresser l'intelligence artificielle (IA) d'un point de vue architectural en proposant un nouveau paradigme appelé architecture adaptative afin de rendre le matériel informatique adaptable à toute application d'IA donnée et à ses contraintes en termes de précision, d'énergie, de latence et de fiabilité. La vision à long terme de cette action est d'aller au-delà de l'état de l'art des architectures et de cibler la prochaine génération d'IA en étudiant et en concevant une IA embarquée flexible, efficace, durable et fiable, et en implémentant sur des architectures adaptatives. Toutes les personnes recrutées pour l'action AdaptING formeront un réseau de collaborations animé périodiquement par tous les partenaires du projet.

Description du projet

Ce projet vise la conception d'architectures de calcul adaptées à l'implémentation d'une vaste gamme d'algorithmes d'intelligence artificielle (IA) et d'apprentissage automatique (AA) (*Machine Learning*) dans l'internet des objets (*Internet of Things* - IoT).

Les algorithmes d'IA et d'AA incluent des opérations d'analyse de données pour différents buts incluant la prédiction, la classification, la segmentation, la préparation de recommandations, et la prise de décisions. Ces algorithmes nécessitent habituellement la manipulation d'une grande quantité d'informations et impliquent un grand nombre de calculs. On s'attend à ce que les résultats produits par ces algorithmes soient corrects et précis, qu'ils soient disponibles au moment nécessaire, et que le traitement utilise le moins d'énergie possible. Ces attentes impliquent souvent que les algorithmes d'IA et d'AA doivent être implémentés dans des architectures de calcul sophistiquées avec des hiérarchies et interfaces mémoires particulières, des caractéristiques qui vont souvent au-delà de celles des microprocesseurs traditionnels.

L'implémentation d'algorithmes d'IA et d'AA dans l'internet des objets (*Internet of Things* - IoT) présente des défis additionnels importants. Les objets sont très variés en nature, en taille, en fonction ou en capacités, et incluent des téléphones intelligents, des voitures autonomes, des caméras intelligentes, des appareils ménagers et des drones miniaturisés. Ils intègrent des capteurs, des unités de traitement, du logiciel, des systèmes de communication et parfois des actionneurs. La nature même des algorithmes d'IA implémentés dans chaque objet varie en fonction de sa tâche et de ses fonctionnalités propres. Dans les cas les plus extrêmes, on peut imaginer des objets qui doivent opérer de manière complètement indépendante pendant des jours, des mois ou des années, et dont la nature de la tâche évolue en fonction du temps, du lieu et des circonstances. On peut donc raisonnablement supposer que ces objets devront être capables d'implémenter une grande variété d'algorithmes d'IA et d'AA pendant leur durée de vie utile.

Il existe plusieurs classes d'algorithmes d'AA : arbres de décision (*Decision Trees*), forêts aléatoires (*Random Forests*), k-plus proches voisins (*K-Nearest Neighbours*), et différentes variantes des réseaux de neurones (de base, convolutionnels, à rétroaction, etc.). Ces algorithmes impliquent des opérateurs de calculs différents, de la nature des opérations réalisées à la précision requise. Chaque calcul peut porter sur un nombre restreint ou important de données qui peuvent être colocalisées en mémoire ou non. Les patrons d'accès à la mémoire peuvent ainsi être très variés. Selon l'application considérée, l'implémentation d'une ou de plusieurs de ces classes d'algorithmes peut être nécessaire. De plus, la nature des calculs et les données utilisées peuvent être significativement différentes dans les phases d'inférence et d'apprentissage du système. Il est difficile pour un processeur unique d'implémenter de façon optimale tous les types d'opérations et d'accès à la mémoire correspondant aux différentes classes d'algorithmes d'AA.

Ce projet porte sur la conception de processeurs pouvant implémenter de façon efficace différents algorithmes d'IA et d'AA pour des objets de l'IoT. Les processeurs devront être flexibles, c'est-à-dire qu'ils devront pouvoir s'adapter aisément et rapidement à différentes classes d'algorithmes. La hiérarchie et les interfaces mémoires du système devront aussi s'adapter facilement à chaque classe d'algorithmes et à leurs représentations et entreposages des données. Afin de soutenir l'opération d'objets autonomes indépendants, les processeurs devront être adaptés aussi bien aux phases d'inférence que d'apprentissage des différents algorithmes. Les processeurs devront par ailleurs rencontrer les spécifications strictes de débit et de latence des domaines d'applications ciblés. Ils devront atteindre une très grande efficacité énergétique de façon à pouvoir être utilisés dans des objets alimentés par batteries.

Objectifs de recherche

Le projet poursuivra les objectifs de recherche suivants.

- Préparer une revue de l’état de l’art pertinent pour le projet.
- Proposer des architectures flexibles permettant d’implémenter plus d’une classe ou sous-classe d’algorithmes d'AA, mettant l’emphase sur le débit ou sur la consommation d’énergie.
- Proposer des hiérarchies et des interfaces mémoires convenant à différentes classes ou sous-classes d’algorithmes d’AA, mettant l’emphase sur le débit ou sur la consommation d’énergie.
- Démontrer la validité des architectures proposées en les implémentant dans des technologies pertinentes (ASIC, FPGA, CGRA, ASIP, etc.).
- Mesurer la performance et la consommation d’énergie des architectures développées dans des contextes d’utilisation réels. Comparer les résultats à l’état de l’art.

Méthodologie

- Revue de la littérature et analyse d’architectures existantes.
    - Dresser une liste de classes d’applications de l’IoT susceptibles de bénéficier de la conception d’une architecture de calcul flexible.
    - Dresser une liste des types d’algorithmes d’IAA et d’AA susceptibles d’être implémentés dans des objets interconnectés. Pour chaque type d’algorithme, déterminer les opérateurs de calculs et les patrons d’accès à la mémoire. Déterminer les invariants pour leur implémentation.
    - Dresser une liste des types d’objets interconnectés selon leur localisation dans le nuage (centre, périphérie, objet embarqué, etc.) et décrire leurs caractéristiques.
    - Dresser un inventaire de l’état de l’art des architectures présentées pour implémenter différentes classes d’algorithmes d'AA (voir entre autres Mourshed 2022, les références citées par Ahmadi 2021, Jouppi 2018, Luo 2017, Chen 2016, Du 2015, etc.).
    - Dresser un inventaire d’architectures de processeurs flexibles, pouvant implémenter efficacement des algorithmes de plusieurs classes différentes, en apprentissage automatique ou dans d’autres catégories.
    - Rédaction d’une revue de littérature.
- Modélisation et implémentation d’architectures existantes choisies et reproduction de résultats.
- Proposition de nouvelles architectures pouvant implémenter les calculs de deux, trois, quatre ou plus classes différentes d’algorithmes. Exploitation entre autres des concepts de parallélisme, de pipeline et de réseaux systoliques, et emphase sur des solutions novatrices pour les interconnexions entre les chemins des données et les mémoires. Modélisation des architectures et estimation de la performance et des coûts.
- Description des nouvelles architectures par diagrammes de flots de données et par langages de description matérielle. Vérification des systèmes par simulation.

- Synthèse et implémentation dans différentes technologies (ASIC, FPGA, CGRA, ASIP, etc.) et extraction des performances atteintes.
- Comparaison des architectures flexibles avec les architectures traditionnelles.
- Rédaction d'articles.

Échéancier

- Octobre 2024 à mars 2025: analyse d'architectures existantes et rédaction d'une revue de littérature
- Mars 2025 à février 2026 : proposition de nouvelles architectures
- Mars 2026 à février 2027 : évaluation des architectures
- Mars à septembre 2027 : rédaction de la thèse et d'articles

Bibliographie

[1] M. Traore, J. M. Pierre Langlois, et J. Pierre David, « ASIP Accelerator for LUT-based Neural Networks Inference », in 2022 20th IEEE Interregional NEWCAS Conference (NEWCAS), juin 2022, p. 524-528. doi: 10.1109/NEWCAS52662.2022.9842211.

[2] M. G. S. Murshed, C. Murphy, D. Hou, N. Khan, G. Ananthanarayanan, et F. Hussain, « Machine Learning at the Network Edge: A Survey », ACM Comput. Surv., vol. 54, nº 8, p. 1-37, nov. 2022, doi: 10.1145/3469029.

[3] M. Ahmadi, S. Vakili, et J. M. P. Langlois, « CARLA: A Convolution Accelerator With a Reconfigurable and Low-Energy Architecture », IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 68, nº 8, p. 3184-3196, août 2021, doi: 10.1109/TCSI.2021.3066967.

[4] M. Ahmadi, S. Vakili, et J. M. P. Langlois, « Heterogeneous Distributed SRAM Configuration for Energy-Efficient Deep CNN Accelerators », in 2020 18th IEEE International New Circuits and Systems Conference (NEWCAS), juin 2020, p. 287-290. doi: 10.1109/NEWCAS49341.2020.9159814.

[5] M. Ahmadi, S. Vakili, et J. M. P. Langlois, « An Energy-Efficient Accelerator Architecture with Serial Accumulation Dataflow for Deep CNNs », in 2020 18th IEEE International New Circuits and Systems Conference (NEWCAS), juin 2020, p. 214-217. doi: 10.1109/NEWCAS49341.2020.9159818.

[6] Y. S. Shao et al., « Simba: Scaling Deep-Learning Inference with Multi-Chip-Module-Based Architecture », in Proceedings of the 52nd Annual IEEE/ACM International Symposium on Microarchitecture, New York, NY, USA, oct. 2019, p. 14-27. doi: 10.1145/3352460.3358302.

[7] I. Palit, Q. Lou, R. Perricone, M. Niemier, et X. S. Hu, « A Uniform Modeling Methodology for Benchmarking DNN Accelerators », in 2019 IEEE/ACM International Conference on Computer-Aided Design (ICCAD), nov. 2019, p. 1-7. doi: 10.1109/ICCAD45719.2019.8942095.

[8] Y.-H. Chen, T.-J. Yang, J. Emer, et V. Sze, « Eyeriss v2: A Flexible Accelerator for Emerging Deep Neural Networks on Mobile Devices ». arXiv, 20 mai 2019. doi: 10.48550/arXiv.1807.07928.

[9] A. M. Abdelsalam, A. Elsheikh, J.-P. David, et J. M. P. Langlois, « POLYCiNN: Multiclass Binary Inference Engine using Convolutional Decision Forests », in 2019 Conference on Design and Architectures for Signal and Image Processing (DASIP), oct. 2019, p. 13-18. doi: 10.1109/DASIP48288.2019.9049176.

[10] X. Liu, W. Wen, X. Qian, H. Li, et Y. Chen, « Neu-NoC: A high-efficient interconnection network for accelerated neuromorphic systems », in 2018 23rd Asia and South Pacific Design Automation Conference (ASP-DAC), janv. 2018, p. 141-146. doi: 10.1109/ASPDAC.2018.8297296.

[11] N. P. Jouppi, C. Young, N. Patil, et D. Patterson, « A domain-specific architecture for deep neural networks », Commun. ACM, vol. 61, nº 9, p. 50-59, août 2018, doi: 10.1145/3154484.

[12] A. M. Abdelsalam, A. Elsheikh, J.-P. David, et J. M. Pierre Langlois, « POLYBiNN: A Scalable and Efficient Combinatorial Inference Engine for Neural Networks on FPGA », in 2018 Conference on Design and Architectures for Signal and Image Processing (DASIP), oct. 2018, p. 19-24. doi: 10.1109/DASIP.2018.8596871.

[13] A. M. Abdelsalam, F. Boulet, G. Demers, J. M. Pierre Langlois, et F. Cheriet, « An Efficient FPGA-based Overlay Inference Architecture for Fully Connected DNNs », in 2018 International Conference on ReConFigurable Computing and FPGAs (ReConFig), déc. 2018, p. 1-6. doi: 10.1109/RECONFIG.2018.8641735.

[14] T. Luo et al., « DaDianNao: A Neural Network Supercomputer », IEEE Trans. Comput., vol. 66, nº 1, p. 73-88, janv. 2017, doi: 10.1109/TC.2016.2574353.

[15] Y.-H. Chen, T. Krishna, J. S. Emer, et V. Sze, « Eyeriss: An Energy-Efficient Reconfigurable Accelerator for Deep Convolutional Neural Networks », IEEE J. Solid-State Circuits, vol. 52, nº 1, p. 127-138, janv. 2017, doi: 10.1109/JSSC.2016.2616357.

[16] Y.-H. Chen, T. Krishna, J. Emer, et V. Sze, « 14.5 Eyeriss: An energy-efficient reconfigurable accelerator for deep convolutional neural networks », in 2016 IEEE International Solid-State Circuits Conference (ISSCC), janv. 2016, p. 262-263. doi: 10.1109/ISSCC.2016.7418007.

[17] Z. Du et al., « ShiDianNao: Shifting vision processing closer to the sensor », in 2015 ACM/IEEE 42nd Annual International Symposium on Computer Architecture (ISCA), juin 2015, p. 92-104. doi: 10.1145/2749469.2750389.

[18] J. Zheng, Y. Liu, X. Liu, L. Liang, D. Chen, et K.-T. Cheng, « ReAAP: A Reconfigurable and Algorithm-Oriented Array Processor With Compiler-Architecture Co-Design », IEEE Transactions on Computers, vol. 71, nº 12, p. 3088-3100, déc. 2022, doi: 10.1109/TC.2022.3213177.