Postdoc: analyse, ordonnancement et déploiement de réseaux de neurones embarqués avec le modèle AER

Claire Pagetti (ONERA), Tomasz Kloda (LAAS), Thomas Carle (IRIT)

Juin 2024

L'ONERA, le LAAS et l'IRIT sont à la recherche d'un.e candidat.e pour un contrat de recherche postdoctorale de 2 ans en lien avec l'implémentation temporellement prédictible de fonctions d'inférences de réseaux de neurones embarqués sur des cibles multi-cœurs.

Introduction

L'introduction progressive d'applications basées sur de l'apprentissage automatique dans des systèmes embarqués temps-réel (par exemples dans les véhicules autonomes) nécessite de développer des méthodes d'analyse et de conception de ces applications qui garantissent que leur exécution respecte des contraintes temporelles strictes. L'une des difficultés provient du fait que ces applications nécessitent à la fois une grande quantité de calculs et de transferts de données, qui imposent l'adoption d'architectures de calcul complexes comme les processeurs multi-cœurs. A leur tour, ces architectures complexifient les techniques d'analyse de pire-temps d'exécution nécessaires à la validation des propriétés temporelles et à la certification du système. En pratique, les différents cœurs qui exécutent des tâches indépendamment les uns des autres se retrouvent en conflit pour accéder à certains composants partagés, tels que les bus mémoire, les caches partagés et les controlleurs de bancs mémoire. Ces conflits ayant un impact fort sur les temps d'exécution des différentes tâches, ils doivent être pris en compte lors du calcul des pire-temps d'exécution. Cependant, ils dépendent de l'entrelacement des accès à la mémoire initiés par les différents cœurs, et du moment où ils sont initiés. Dans une approche pire-cas, considérer cette interférence entre les cœurs ajoute un niveau de pessimisme aux résultats de l'analyse qui n'est pas acceptable dans un contexte industriel.

Le modèle Acquisition-Exécution-Restitution [4] (AER, aussi connu sous le nom REW ou 3-phases) a été développé dans le but de faciliter l'analyse temporelle sur les cibles multi-cœurs. Dans ce modèle, le code de chaque tâche est compilé de façon à ce que son exécution se déroule en 3 phases successives et bien déterminées. L'exécution débute par une phase d'acquisition, au cours de laquelle le code et les données nécessaires à l'exécution de la tâche sont chargés depuis la mémoire partagée vers une mémoire privée (un cache L1 ou un scratchpad) au cœur qui va exécuter la tâche. La tâche s'exécute ensuite sur le cœur, en accédant uniquement à la mémoire privée. Une fois la tâche exécutée, une phase de restitution copie les résultats calculés par la tâche dans la mémoire partagée. En veillant à ordonnancer les phases d'acquisition et de restitution des différentes tâches de manière à ce qu'elles se produisent toujours séquentiellement, il est possible de supprimer toute interférence dans le système, ce qui simplifie l'analyse et retire une quantité significative de pessimisme.

D'un autre côté, l'implémentation efficace de fonctions d'inférence de réseaux de neurones nécessite d'optimiser l'utilisation des différents niveaux de caches présents dans les cibles matérielles. Cela se traduit par des algorithmes réalisant des transferts de données depuis la mémoire partagée vers des caches privés et/ou partagés, puis des phases de calcul pour lesquelles les données sont accédées directement depuis les caches [1].

Il apparait ainsi une certaine compatibilité entre le modèle AER et la manière dont les fonctions d'inférence sont implémentées et exécutées. Dans cette optique, il s'agira dans ce postdoc de pousser cette logique jusqu'au bout, et de concevoir un générateur de code AER pour les fonctions d'inférence.

Description du poste

L'objectif du postdoc est d'étendre les travaux réalisés sur le générateur automatique de code d'inférence ACETONE [2, 3] développé à l'ONERA. Il s'agit en particulier :

- d'intégrer les derniers résultats concernant l'implémentation de fonctions optimisées pour la multiplication de matrices [1] dans ACETONE.
- d'intégrer le modèle AER dans le code généré par ACETONE : séparer les fonctions en phases d'acquisition, exécution et restitution.
- d'intégrer un algorithme d'ordonnancement compatible AER pour des cibles multi-cœurs et de générer automatiquement le code correspondant ainsi que les synchronisations nécessaires.
- de réaliser l'analyse de pire-temps d'exécution sur le code généré.
- de valider expérimentalement l'approche sur une série de benchmarks.
- (si le temps le permet) de développer un démonstrateur basé sur un cas d'étude de véhicule autonome (e.g. modèle réduit de voiture autonome ou drone volant).

Le postdoc sera réalisé à l'IRIT au sein de l'équipe TRACES, et co-encadré à l'ONERA et au LAAS. Le financement de 2 ans est tiré d'un projet ANR et le postdoc sera également rattaché à la chaire ANITI "embarquabilité de l'IA". Nous visons des publications dans les conférences top-tier de la communauté temps-réel (RTSS, RTAS, ECRTS) et dans des journaux Q1 et Q2 (IEEE TC, RTS, ACM TECS).

Compétences demandées

En plus d'avoir un doctorat en informatique, la/le candidat.e devra faire preuve d'initiative et d'indépendance, et être compétent.e dans un ou plusieurs des domaines suivants :

- Systèmes temps-réel, ordonnancement
- Programmation C, assembleur, python
- Analyse statique, compilation
- Notions de machine learning, réseaux de neurones
- Rédaction d'articles scientifiques

Contacts

Pour candidater, merci d'envoyer un CV et une lettre de motivation à thomas.carle@irit.fr, claire.pagetti@onera.fr et tkloda@laas.fr

Références

[1] I. De Albuquerque Silva, T. Carle, A. Gauffriau, V. Jegu, and C. Pagetti. A predictable simd library for gemm routines. In *30th Real-Time and Embedded Technology and Applications Symposium, RTAS 2024, May 13-16, 2024, Hong Kong, China*, 2024.

[2] I. De Albuquerque Silva, T. Carle, A. Gauffriau, and C. Pagetti. ACETONE : predictable programming framework for ML applications in safety-critical systems. In *34th Euromicro Conference on Real-Time Systems, ECRTS 2022, July 5-8, 2022, Modena, Italy*.

[3] I. De Albuquerque Silva, T. Carle, A. Gauffriau, and C. Pagetti. Extending a predictable machine learning framework with efficient gemm-based convolution routines. *Real Time Syst.*, 59(3) :408–437, 2023.

[4] G. Durrieu, M. Faugère, S. Girbal, D. Gracia Pérez, C. Pagetti, and W. Puffitsch. Predictable flight management system implementation on a multicore processor. In *ERTS'14*, 2014.