Tremplin Recherche

Déploiement des applications IA temps réel sur des architectures hétérogènes:
Modèle de tâche pour les réseaux DNN Temps réel

Laboratoire d'accueil : LIGM,

Encadrements : Mourad DRIDI, Yasmina ABDEDDAÏM
mourad.dridi@esiee.fr; yasmina.abdeddaim@esiee.fr

Partenaire international : Italie
Université de Modène et de Reggio d'Émilie / Université de Turin.

Filière concernée : Systèmes embarqués

Mots clés : Systèmes embarqués temps réel, DAG task, CPU, GPU, Modèle de tâche

1. CONTEXTE

Le déploiement des applications temps réel, utilisant des fonctionnalités d'Intelligence Artificielle (IA) (lors de la phase d'inférence) sur la même plate-forme d'exécution nécessite une puissance de calcul élevée, qui ne peut être satisfaite aujourd'hui que par des plateformes hétérogènes, combinant CPUs et accélérateurs (GPU).

Une **architecture de calcul hétérogène** distribue les données, le traitement et l'exécution des programmes entre les différentes unités de calcul qui sont les mieux adaptées aux tâches spécifiques. Ces systèmes permettent de gagner en performances de calcul mais introduisent également de la complexité dans le développement des systèmes, la gestion des ressources, les protocoles de communications entre les différentes unités de calcul [1].

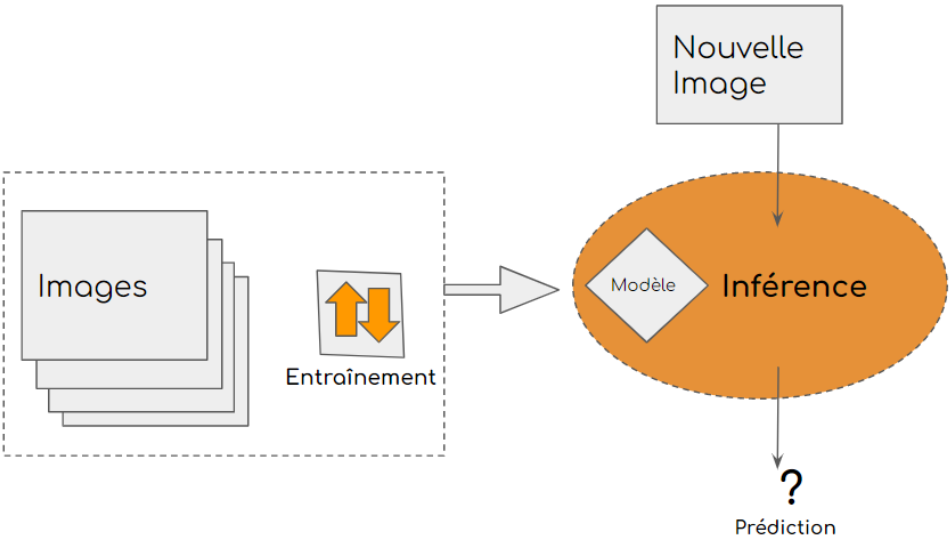
Les principaux fournisseurs de matériel tels que NVIDIA, Qualcomm ou AMD proposent plusieurs plates-formes COTS hétérogènes. NVIDIA Jetson AGX est un exemple de ces architectures. Cette architecture embarque 8 cœurs ARM, 1 GPU, deux DLA (Deep Learning Accélérateurs) et un PVA (accélérateur de vision) [1,2].

Le fonctionnement de nombreux algorithmes d'apprentissage machine est basé sur deux phases de traitement distinctes : l'entraînement et l'inférence. La phase d'entraînement fonctionne sur un large ensemble de données pour en tirer des enseignements. La phase d'inférence ou de prédiction consiste à prédire un résultat sur des nouvelles données en utilisant les enseignements de la phase d'apprentissage [2]. Les architectures hétérogènes

peuvent permettre aux systèmes embarqués d'utiliser plus efficacement des algorithmes d'IA et cela en profitant des ressources matérielles qu'elles offrent pour effectuer l'inférence des données localement.

Exemple d'application : Self-Driving Car

Les véhicules autonomes utilisent les réseaux DNN temps réel afin de détecter les objets physiques et les marquages au sol en analysant les images produites par plusieurs caméras montées à l'avant, sur le côté et à l'arrière de la voiture.



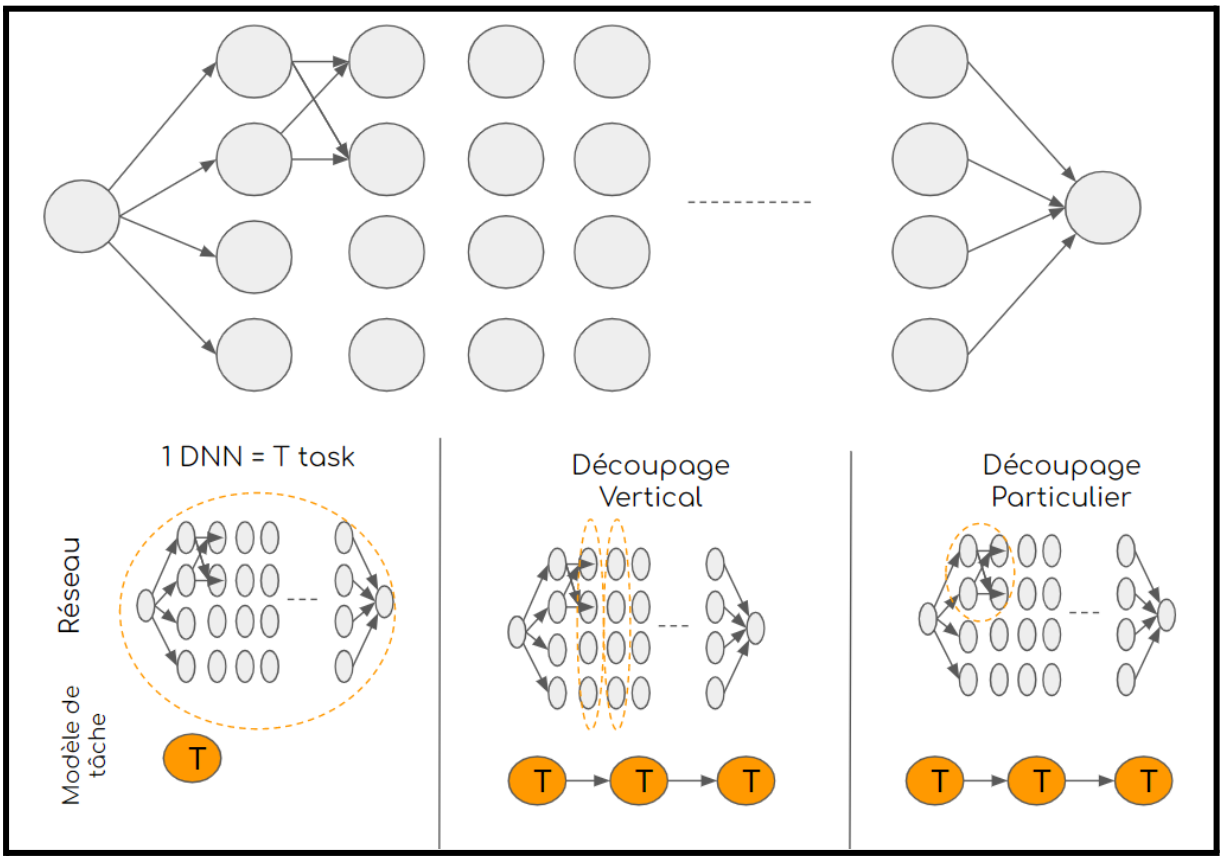
2. PROBLÉMATIQUE

Les frameworks DNN existants, tels que Caffe, TensorFlow et Torch, ne permettent d'affecter qu'une seule priorité pour chaque DNN et inférence séquentielle ce qui peut être particulièrement problématiques dans le contexte de déploiement de plusieurs DNN temps réel [6, 7]. En effet, cette politique d'affectation de priorités conduit à un surdimensionnement du système qui vient de la sous exploitation du parallélisme qui peut être offert par les architectures hétérogènes.

3. OBJECTIFS

Actuellement, les modèles de tâches classiques ne considèrent pas les spécifications des réseaux de neurones (DNN temps réel) [6]. L'objectif de ce projet est de proposer un modèle de tâche pour DNN temps réel qui favorise le parallélisme entre les tâches et améliore l'utilisation de ressources (CPU/GPU)

Ce modèle doit respecter les contraintes des connexions entre les nœuds qui caractérisent les DNN temps réel.



Références

[1] H. Andrade and I. Crnkovic, "A Review on Software Architectures for Heterogeneous Platforms," 2018 25th Asia-Pacific Software Engineering Conference (APSEC), Nara, Japan, 2018, pp. 209-218, doi: 10.1109/APSEC.2018.00035.

[2] H. Zhou, S. Bateni and C. Liu, "S3DNN: Supervised Streaming and Scheduling for GPU-Accelerated Real-Time DNN Workloads," 2018 IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS), 2018, pp. 190-201, doi: 10.1109/RTAS.2018.00028.

[3] Mourad Dridi, Frank Singhoff, Stéphane Rubini, Jean-Philippe Diguet: ECTM: A network-on-chip communication model to combine task and message schedulability analysis. J. Syst. Archit. 114: 101931 (2021)

[4] Mourad Dridi, Stéphane Rubini, Mounir Lallali, Martha Johanna Sepúlveda Flórez, Frank Singhoff, JeanPhilippe Diguet: Design and Multi-Abstraction-Level Evaluation of a NoC Router for Mixed-Criticality Real-Time Systems. ACM J. Emerg. Technol. Comput. Syst. 15(1): 2:1-2:37 (2019)

[5] Mourad Dridi, Stéphane Rubini, Mounir Lallali, Martha Johanna Sepúlveda Flórez, Frank Singhoff, JeanPhilippe Diguet: DAS: An Efficient NoC Router for Mixed-Criticality Real-Time Systems. ICCD 2017: 229-232

[6] S. K. Roy, R. Devaraj and A. Sarkar, "SAFLA: Scheduling Multiple Real-Time Periodic Task Graphs on Heterogeneous Systems," in *IEEE Transactions on Computers*, 2022, doi: 10.1109/TC.2022.3191970.

[7] Houssam-Eddine Zahaf, Nicola Capodieci, Roberto Cavicchioli, Giuseppe Lipari, Marko Bertogna: The HPC-DAG Task Model for Heterogeneous Real-Time Systems. IEEE Trans. Computers 70(10): 1747-1761 (2021)