PHD PROPOSAL IN ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

Certified programming framework for Machine Learning applications

Advisor (s): Claire Pagetti and Aurélien Plyer (ONERA) – name.surname@onera.fr
Adrien Gauffriau (Airbus)

https://www.onera.fr/fr/staff/claire-pagetti

Net salary: 2096 € per month with some teaching (64 hours per year on average)

Duration: 36 months

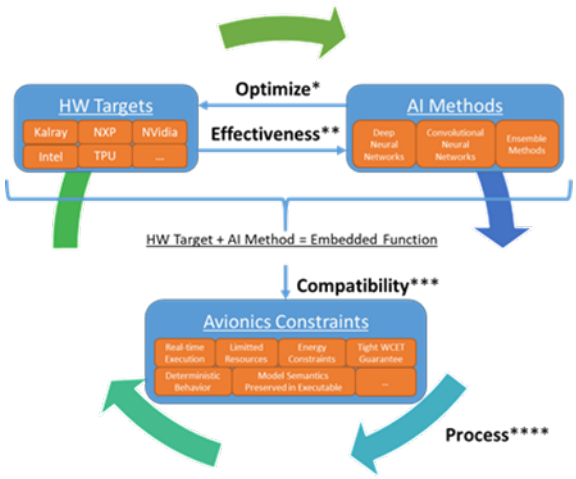
CONTEXT

Machine Learning gains an important consideration in the domain of safety critical systems, including aeronautical area. However, as those applications do not reach classical safety confidence levels and are not implemented with accepted development process [BCM+15, ABH+18], many research and engineering activities must be conducted before embedding these kinds of application in aircrafts. Among them, the question of how to safely and reliably implement a neural network on an adequate hardware is of vital importance. Indeed, certification requirements, in particular those of the DO 178C [RTC11], impose strong guarantees on the quality of the code and expect the designer to compute the WCET (Worst Case Execution Time).

OBJECTIVES

The scope of the PhD is the real-time implementation of neural networks on platforms. Thus, the purpose of the PhD is to answer the following questions:

- how to choose an adequate COTS (Commercial Off the Shelf) hardware that offers sufficient computer performance and that fulfills aeronautical constraints (e.g. dissipation); which COTS is an adequate choice for which type of ML method
- how to define an execution model [PMN+16] on the hardware so that it is possible to compute tight WCET of a Machine Learning model/applications
- how to code efficiently a Machine Learning model and compile it on the target
- how to parallelize the execution if the platform offers parallelism



* **Optimize** AI model for HW Platform: existing frameworks, custom implementation

** **Evaluate** HW Targets **Effectiveness** & Efficiency for Various AI Methods

*** **Assess Compability** of the HW Target & AI Method combinations **with Avionics Constraints**

**** Propose a **complete production process** (from AI model to embedded function) fully compatible with avionics constraints (including model semantics preservation)

PHD PROGRESS

First of all, aeronautical inputs will be clarified. The PhD will be in charge to investigate Airbus needs and future aircraft systems. The study will be restrained to the so called supervised learning (for instance no reinforcement learning), that is the embedded function is for inference only (model training is realized offline on ground in specialized frameworks). The families of machine learning will belong to Deep Neural Networks, Ensemble Methods and Convolutional Neural Networks.

The first objective is to review existing COTS technologies available on the market to execute neural network applications. A non exhaustive preliminary list is: NXP QorIq family (e.g. T1042 [NXP15]), Kalray Coolidge [Kal19], NVIDIA GPU (e.g. Turing [NVI18]), Intel Movidius Neural Compute Stick [Int18] or TPU (e.g. Coral Dev Board [Goo19]). A design space exploration will be realized on those platforms with the use cases. In addition to the pure hardware part, it is also important to investigate the associated frameworks (e.g. TensorFlow), the available code generation approaches and compilation procedures of these frameworks. Their export formats is of great investigation interest, since these are the entry points to the embedded function generation phase. In addition to these technical considerations, the investigation will be made also in regards to the certification and industrial constraints.

From the exploration, one or two COTS will be chosen. The second objective is to define for the candidates a so called execution model, which is a set of rules to program and configure the platform in order to reduce non predictable behavior. In the real-time community, predictability is the ability to compute tight WCET. Recent generation of processors embeds highly complex, and often non documented, mechanisms making it hard to assess the maximal number of cycles required to execute a sequential program [WEE+08]. One solution to overcome this problem is to reduce the potential non predictable behavior by restraining the execution according to pre-defined programming rules (e.g. TDMA time driven multiple access to a shared resource). The execution will be done on a minimalist environment, e.g. bare metal, to fully understand and predict the low level behavior of the program and the corresponding computational resources requirement.

Once the target and its associated execution model have been defined, the last part of the PhD will consist in developing an automatic, and verified, framework to generate low level code (e.g. C code) and its associated binaries on the target. By verified, we mean that the semantics of the

C code must be equivalent to the output of the neural network framework design (e.g. TensorFlow or PyTorch); and the semantics of the execution must be equivalent to the code.

Understand the efficiency of various hardware architectures for different Machine Learning methods as well as to evaluate our capability to embed these methods on these platforms with avionics constraints in mind.

REFERENCES

[ASK+18] Tahmid Abtahi, Colin Shea, Amey Kulkarni, and Tinoosh Mohsenin. Accelerating Convolutional Neural Network With FFT on Embedded Hardware. IEEE transactions on very large scale integration (VLSI) systems. 2018.

[ABH+18] E. Alves, D. Bhatt, B. Hall, K. Driscoll, A. Murugesan, et J. Rushby, « Considerations in Assuring Safety of Increasingly Autonomous systems », NASA, NASA/CR-2018-220080, 2018.

[BCM+15] S. Bhattacharyya, D. Cofer, D. J. Musliner, J. Mueller et E. Engstrom, « Certification considerations for adaptive systems », NASA, NASA/CR-2015-218702, 2015.

[Goo19] Google. Coral Dev Board. 2019 https://coral.withgoogle.com/docs/dev-board/datasheet/

[Int18] Intel. Intel Movidius Neural Compute Stick. https://software.intel.com/en-us/neural-compute-stick

[Joo13] Mohammad Hadi Jooybar. Deterministic Execution on GPU Architectures. Master thesis. 2013.

[Kal19] Kalray. MPPA-3 – Coolidge. 2019. https://www.kalrayinc.com/release-of-third-generation-mppa-processor-coolidge/

[NVI18] NVIDIA Emmett Kilgariff, Henry Moreton, Nick Stam and Brandon Bell. Turing. 2018. https://devblogs.nvidia.com/nvidia-turing-architecture-in-depth/

[NXP15] Freescale Semiconductor. QorIQ T1042, T1022 Data Sheet. 2015. https://4donline.ihs.com/images/VipMasterIC/IC/PHGL/PHGL-S-A0002440345/PHGL-S-A0002440345-1.pdf

[PMN+16] Quentin Perret, Pascal Maurère, Eric Noulard, Claire Pagetti, Pascal Sainrat, Benoit Triquet.,Temporal Isolation of Hard Real-Time Applications on Many-Core Processors. RTAS 2016: 37-47

[RTC11] RTCA, Inc. DO-178 ED-12C - Software Considerations in Airborne Systems and Equipment Certification, 2011.

[WEE+08] R. Wilhelm, J. Engblom, A. Ermedahl, N. Holsti, S. Thesing, D. Whalley, G. Bernat, C. Ferdinand, R. Heckmann, T. Mitra, F. Mueller, I. Puaut, P. Puschner, J. Staschulat, and P. Stenstrom. The worst-case execution-time problem - overview of methods and survey of tools. ACM Transactions Embedded Computing Systems, 7(3):36:1–36:53, May 2008.

APPLICATION PROCEDURE

Formal applications should include detailed cv, a motivation letter and transcripts of bachelors' degrees.
Samples of published research by the candidate and reference letters will be a plus.
> applications should be sent by email to: advisor email
More information about ANITI: https://aniti.univ-toulouse.fr/