POLITECNICO DI TORINO
Dipartimento di Automatica e Informatica

PhD in Computer and Control Engineering
XXXII cycle

Supervisor
Prof. Andrea Acquaviva
Prof. Enrico Macii

Many-core and heterogeneous architectures: programming models and compilation toolchains

PhD Candidate: *Francesco Barchi*

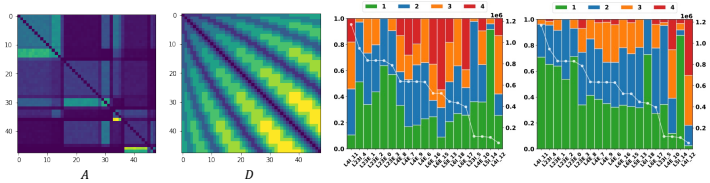
1. Introduction

In heterogeneous and many-core platforms, the mapping procedure of "what to do" (computing, communication, storage) and "where to do it" is not a trivial task. In this scenario, my research work evolved from many-core platforms in the neuromorphic field, through the development of middleware to support parallel programming models, to reach machine-learning techniques for IR code classification and domain specific compilation pipelines for ultra low power heterogeneous systems.

2. SpiNNaker – SNN Mapping

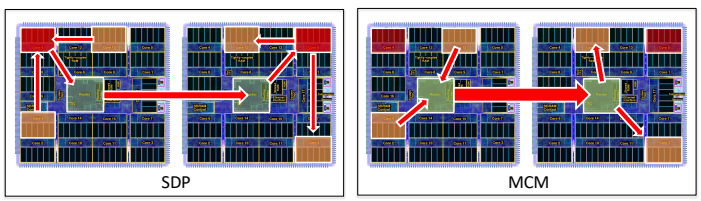
The SNN mapping in SpiNNaker is relevant; an inefficient allocation impacts the reliability of the network execution. We formalised the problem by breaking it into two phases: i) *Graph Partitioning*, the network is divided into subsets satisfying two constraints: neuron model type and computational complexity. ii) *Graph Placement*, formalised as a permutation search problem using: A (SNN adjacency matrix), D (distance matrix of SpiNNaker nodes), π (permutation vector), and the function f (synaptic elongation) to minimise. Using a basin-hopping optimisation technique, we obtained a 25% improvement in f [3,4].

$$f = e^T(A \odot D_\pi)\, e = \sum_i \sum_j a_{ij}\, d^\pi_{ij} \qquad D_\pi = P_\pi D P_\pi$$



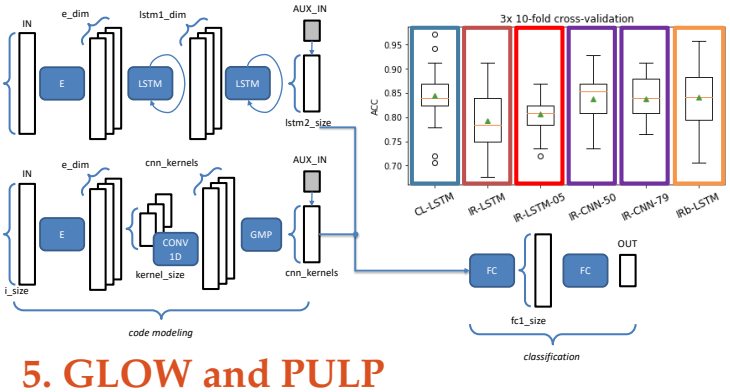
3. SpiNNaker – MPI

The goal of the MPI software stack is provided to SpiNNaker a parallel programming model based on message passing, exploiting its connectivity. To overcome the hardware limitation of the architecture when is required a p2p communication, we have developed a middleware (MCM) capable of diverting unicast, broadcast and synchronization communications on the multicast network. The MPI implementation was built on top of MCM, and virtual Memory Entities, a mechanism to share memory regions around the 768 processors of a SpiNNaker Board [2,5].



4. LLVM-IR Code Classifier

The goal is to choose the fastest compute unit in a heterogeneous target platform for a specific pair of code and data. The approach works at the LLVM-IR level. I use an LLVM front-end compiler (e.g. Clang) to perform the compilation of an application expressed in a high-level programming language. Using an intermediate code representation, the classifier is independent of the source language and the underlying hardware. The language modelling part is based on an LSTM (Long Short-Term Memory) or CNN (1D Convolutional) network, and the classifier part is a sequence of dense layers. We obtained 84% of mean classification accuracy. [1].



5. GLOW and PULP

I explored state of the art in the field of domain-specific compilers able to optimize the execution of computational graphs (Machine Learning, Linear Algebra, Stencil Computations) studying the most prominent actors in this field: MLIR from Google, GLOW from Facebook and TVM. I started to develop a RI5CY backend for GLOW, a runtime for the network model manager in a simple PULP platform and the helper tool XPulp (cross-pulp) for simplifying cross-toolchain compilation. I performed a preliminary study to introduce new GLOW optimization steps for PULP architectures.

6. References

[1] "Code Mapping in Heterogeneous Platforms Using Deep Learning and LLVM-IR". 56th ACM/ESDA/IEEE Design Automation Conference (DAC), 2019.
[2] "Flexible On-line Reconfiguration of Multi-core Neuromorphic Platforms". IEEE Transactions on Emerging Topics in Computing, 2019.
[3] "Mapping Spiking Neural Networks on Multi-core Neuromorphic Platforms: Problem Formulation and Performance Analysis". IFIP Advances in Information and Communication Technology, 2019
[4] "Directed Graph Placement for SNN simulation into a multi-core GALS architecture ". 26th IFIP/IEEE International Conference on Very Large Scale Integration, VLSI-SoC 2018
[5] "An Efficient MPI Implementation for Multi-Core Neuromorphic Platforms". 1st New Generation of Circuits and Systems, NGCAS 2017

All references have as authors: F.Barchi et al.