Progress report (TODO)

Matthias Ehrlich Sebastian Jeltsch Eric Müller
Sven Schrader Bernhard Vogginger

January 18, 2012

WP 3 Task 2: Software development for the HMF

Enabling users to efficiently emulate large neuronal network models spanning over multiple wafers on the HMF poses several new software challenges. The full set of processes that will control the neuromorphic part of the HMF resembles the task of an operating system (OS) that controls the hardware of a common PC. Its workflow is depicted in (Figure 1) and has been developed in close collaboration with BrainScaleS' hardware-, software-, and modeling groups.

An OS for the HMF

The major requirements for the new software framework are: (1) computing valid hardware configurations that represent the user's model. (2) handling runtime data (mostly spikes) at high bandwidth and (3) providing many users in parallel an interface that fits their workflows. In addition, users should be able to use a common neuronal network description language like *PyNN* without the need to be concerned about hardware specifics. In the following, a software framework is presented that tackles these challenges.

At the highest level, the user interacts with the *PyNN* interface. Below *PyNN*, a thin adapter layer called *pyHMF* translates between Python and the next (`C++`-based) layer. This (cf. Figure 1) so-called *Euter* layer serializes the *PyNN*-based network description into a binary format. *PyNN*'s `end()` function initiates the transfer to the *Ester proxy* (some other *PyNN* functions might also trigger data transfers to and from the server). Apart from handing down the network description to the mapping process, the *Ester proxy* also arbitrates user requests, for example it may block experiments if a maximum number of pending jobs is reached.

The biological domain of *PyNN*-based descriptions (e.g. time, parameters, topological structure) has to be translated into the domain of the neuromorphic hardware. This *mapping* (which includes placement and parameter translation of neurons and routing in terms of topological resources) can be rather *large* in terms of computation and memory consumption, depending on the network size. Therefore, the existing mapping

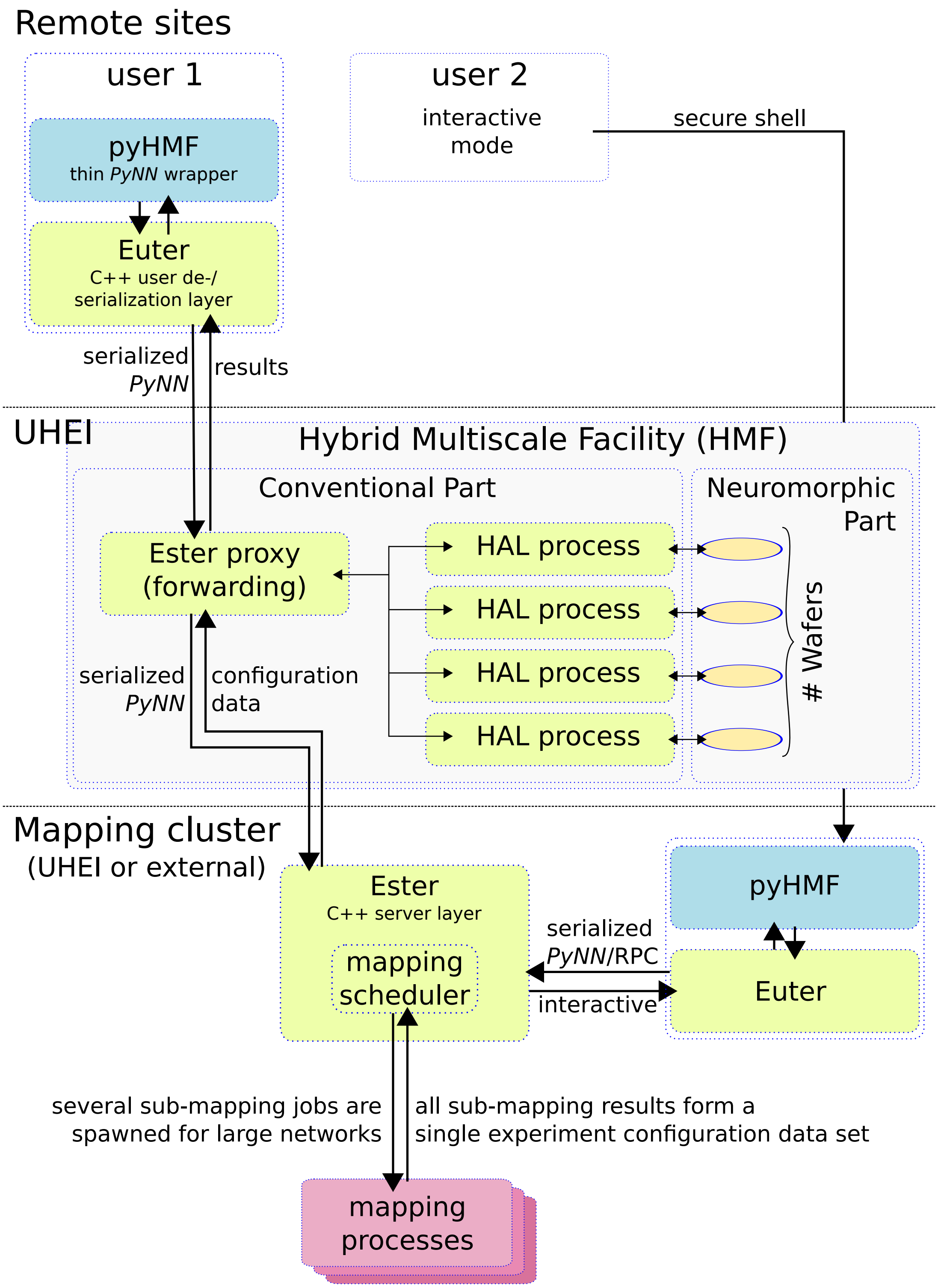

Figure 1: Overview of the new testbed for a scalable software framework.

framework, called *MappingTool*, is being extended to enable a parallel mapping process (cf. WP3 Task 3).

The workload of the HMF is further increased by enabling batch processing of experiments. This is handled by *Ester* (cf. Figure 1) which manages concurrent requests (i.e. mapping jobs) of users and schedules the creation of distributed mapping jobs on the mapping cluster. The current throughput (user requests per time unit) is remarkably large – *Ester* alone could in principle manage thousands of jobs simultaneously.

Interactive mapping and experiment execution is provided via a secure shell-based access. In this case the user-side software stack runs directly on the mapping cluster.

Executable System Specification

The Executable System Specification (*ESS*) is a detailed software model of the BrainScaleS neuromorphic wafer-scale hardware implemented in C++. Its purpose is two-fold. On the one hand it serves as a testbench for the software stack of the BrainScaleS hardware, on the other hand it allows to explore the capabilities and constraints of the BrainScaleS neuromorphic hardware by running neural network simulations on its virtual version whilst the physical model is not yet available.

The initial version of the ESS was initially developed during the BrainScaleS predecessor project FACETS and has been enhanced in several aspects, mostly to be in correspondence with the real wafer-scale system. In particular, the on-wafer routing network has been successfully adapted to the hardware design. Several functional units have been added or further enhanced in the software model of the HICANN building block. Among that are the background event generators, priority encoders. They have been implemented in a clocked fashion, which introduces hardware-specific distortions to the sequence of spikes, e.g. in case of spike bursts some spikes are delayed or even lost. In the latter case, those lost events are recorded and can be evaluated at the end of simulation.

The ESS is now publicly available on the web as part of a Linux live CD. This includes the whole software flow from *PyNN* over the mapping process to the experiment conduction using the ESS and vice-versa. Thus, the ESS is now reflecting the hardware system in terms of functionality, configurability and constraints except that *Spike Timing Dependent Plasticity* (STDP) and hardware distortions (e.g. transistor mismatches) from chip manufacturing remain to be implemented.

During the "*ESS + Neuromorphic Hardware Workshop*" from October, 4th to 6th 2011 at the *Technische Universität Dresden* 25 participants form several direct and indirect project collaborators made their first successful attempts in running neural network experiments with the ESS. Attendees could not only gain a first insight into the BrainScaleS neuromorphic hardware and its behavior, but also give valuable feedback about the software operating the wafer-scale hardware.

The main focus of the next year will be the parallelization of the ESS. Additional efforts will be put into the implementation of STDP and the distribution of the ESS to a larger community, not only as a live system but also as an open source software package.

WP 3 Task 3: Mapping models of biological networks onto the HMF

The mapping process enables modelers and non-hardware experts to deploy their abstract descriptions of neural networks on the neuromorphic component of the HMF. Due to the fact that approximately $N!$ (with N being the number of neurons) possibilities exist for assigning abstract neurons to hardware neurons this becomes more and more difficult as the numbers of neurons increase. It turns out that this problem is equivalent to a graph isomorphism problem, which has been proven to be NP-complete from a computational point of view.

Since the original implementation within the FACETS project, many improvements have been carried out [Brüderle et al., 2011]. The user interface has been updated to match the current upstream version of *PyNN* (`http://neuralensemble.org/trac/PyNN`), the internal hardware representation has been extended to match the actual wafer-scale systems topology and its hardware parameter ranges. Most importantly, however, a new, faster and more flexible routing algorithm has been developed and integrated into the mapping flow. Now, connections can be routed around unavoidable defects on a wafer which is crucial for its real world application.

Much work has been done towards an operating system for neuromorphic hardware. One of the main tasks of classical operating systems is to hide hardware details from the user by providing some layer of abstraction. The mapping process behaves in a very similar fashion for the wafer-scale system.

To improve user experience, run times and flexibility in terms of different and evolving hardware systems, many changes will be conducted during the project. The key aspects for accelerating the process are to use more of the preexisting hierarchical information provided by the network description, switch to a consistent pipeline architecture and to distribute computation beyond single machine boundaries in a distributed fashion.

The flexibility will enhanced by introducing a generic description language for neuromorphic hardware, an effort which has been started in late 2011. This development will be pursued in the future in cooperation with other research groups working on neuromorphic hardware devices.

WP 3 Task 4: Operation of the HMF and Web access to the facility

Operation of the HMF

Network models running on the ESS

The simulation environment ESS enables future HMF-users to develop and test their models before the hardware is operational (“Phase 1”). Currently, several models are in development in collaboration with other WP members according to the BrainScaleS proposal. A large number of these models have been started at the “*ESS + Neuromorphic*

Hardware Workshop" in Dresden (see above). As of now, six network models have been implemented in *PyNN*, most are already functional on the ESS, meaning that they are ready to be tested on the actual hardware. Both the models and the ESS have been benefiting from each other during development, as each "side" has been kept informed about the requirements from the other. This interaction will be further pursued in the following year.

HMF – Conventional Part

The *Hybrid Multiscale Facility* has two major parts: an intra-connected multi-wafer neuromorphic system (cf. WP3 Task 1) and a cluster comprising 16 nodes. The latter, "Conventional Part", needs to handle an enormous data throughput as the expected data volume (from and to the wafers) is very large. The nodes are equipped with an Intel Core i7-2600 CPU, 16 GB of RAM and a remote DMA-capable network interface card providing access to the 10 GbE-based network. All nodes boot from network. Actually, only 6 nodes contain local (solid state) disks, making them capable of dumping one wafer communication channel to disk each.

Inter-connectivity will be provided via a 24-port 10 GbE-switch using low-latency `SFP+ direct attach` cables. Data communication to and from a wafer is handled by 12 FPGA-boards that feature one (in the current setup) Gigabit-Ethernet uplink. These 12 Gigabit-Ethernet links per wafer are aggregated into a single 10 GbE link (per wafer) using an aggregation switch with itself is hooked to the 10 GbE-switch. Preliminary measurements show a round-trip latency of down to 10 µs at 2048 byte packet size. This equals approximately 500 MB/s.

References

D. Brüderle, M.A. Petrovici, B. Vogginger, M. Ehrlich, T. Pfeil, S. Millner, A. Grübl, K. Wendt, E. Müller, M.O. Schwartz, D. Husmann de Oliveira, S. Jeltsch, J. Fieres, M. Schilling, P. Müller, O. Breitwieser, V. Petkov, L. Muller, A.P. Davison, P. Krishnamurthy, J. Kremkow, M. Lundqvist, E. Muller, J. Partzsch, S. Scholze, L. Zühl, C. Mayr, A. Destexhe, M. Diesmann, T.C. Potjans, A. Lansner, R. Schüffny, J. Schemmel, and K. Meier. A comprehensive workflow for general-purpose neural modeling with highly configurable neuromorphic hardware systems. *Biological Cybernetics*, 104 (4):263–296, 2011.