PERPLEXUS: A SIMULATION ENVIRONMENT APPLIED TO LARGE SCALE SPIKING NEURAL NETWORKS

Javier Iglesias [1], Vladyslav Shaposhnyk [1], Olivier Brousse [2], Thierry Gil [2], François Grize [3], Gilles Sassatelli [2] *and* Alessandro E.P. Villa [1]

(1) Grenoble Institute of Neuroscience, Université Jospeh Fourier, Grenoble, France;
(2) Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier, Université Monpellier 2, CNRS, Montpellier, France;
(3) Institut des Systèmes d'Information, Université de Lausanne, Lausanne, Suisse

INTRODUCTION

The goal of the Perplexus Project [1] is to develop a scalable hardware made of custom reconfigurable devices. The platform will enable the simulation of complex systems and the study of emergent behaviours in a wireless network of ubiquitous modules. These modules will act as a distributed sensor capable of computing parts of a simulation using environmental parameters.

A multi-agent platform based on the Java Agent Development Framework (JADE) was proposed to create distributed applications suiting mobile interacting nodes in an evolving environment. The software framework leverages the bio-inspired features of the underlying hardware and sustains philogenetic, ontogenetic and epigenetic processes.

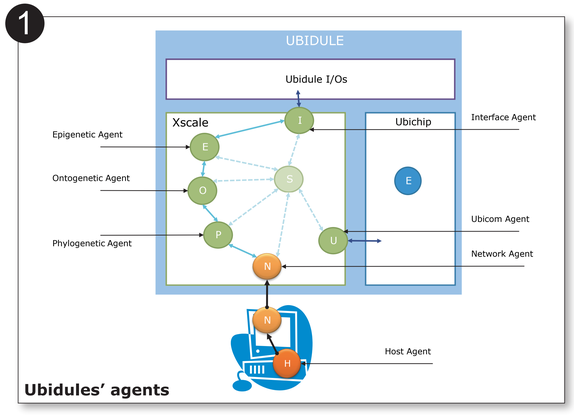
NEURAL MODEL

We applied the initial development tools currently available from the Project to simulate a neural network characterized by excitatory and inhibitory populations of spiking neurons that are initially connected according to probability density functions [2]. The inhibitory population have sparse local connectivity, whereas the excitatory neurons have dense local connectivity and a very sparse but flat probability to send projections anywhere in the network.

The excitatory-excitatory and excitatory-inhibitory synapses are modifiable and take discrete strength values that can change according to a spike-timing dependent plasticity (STDP) rule. We assume a background noise of 5 spikes/s during an initial developmental phase featuring cell death associated with an excessive firing rate, somewhat akin to the neurotoxic effect of glutamate. This phase is followed by the onset of STDP and by recursive spatially and temporally organized stimuli. We have shown that this network dynamics is able to generate detectable preferred firing sequences.

See poster 143.6. (poster board A39) for details

1



Ubidules' agents

FRAMEWORK ARCHITECTURE

Each computational module is an hybrid system that consists of XScale and UbiChip processors at the hardware level (see below) and GNU/Linux and JADE platforms at the software level.

Hardware and software platforms are structured according to three axes:

Phylogenesis based on the evolution of species which leads to adaptation to the environment;

Ontogenesis describing the development of the organisms during early stages;

Epigenesis refering to adaptation, learning and plasticity processes.

Interoperability among the hardware and the software parts of the framework is provided by the use of this bio-inspired structure.

The neural network simulations are logically divided into appropriate software agents: P, O and E corresponding to the 3 axes described above. Infrastructure agents are also provided to handle the input/output (I) sensors/actuators – including cameras, microphones and motors – communications with networks (N) and ubichips (U) and spying (S) (see Figure 1).

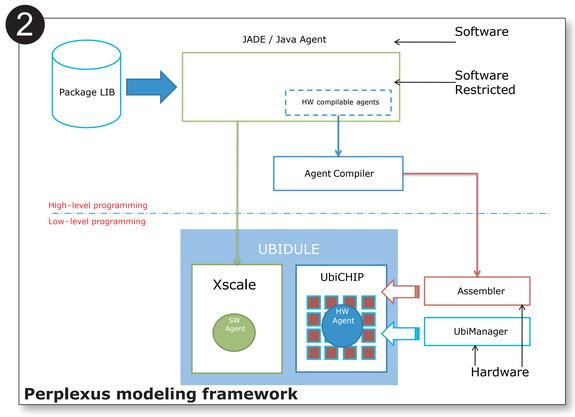
HARDWARE PLATFORM

The Ubidule is the main component of the Perplexus architecture. It features an XScale processor running the GNU/Linux operating system (OS) and allows the connection of USB devices. The programming and management of the platform is facilitated by the numerous tools available for the Free OS, while the basic features can be extended through the use of application-specific sensors and actuators.

Besides the main CPU, a custom reconfigurable device named Ubichip is being developed. It can be configured either in native mode (FPGA-like) or in multiprocessor mode on which epigenetic processes, like neural network simulation, can be mapped on an array of processing elements.

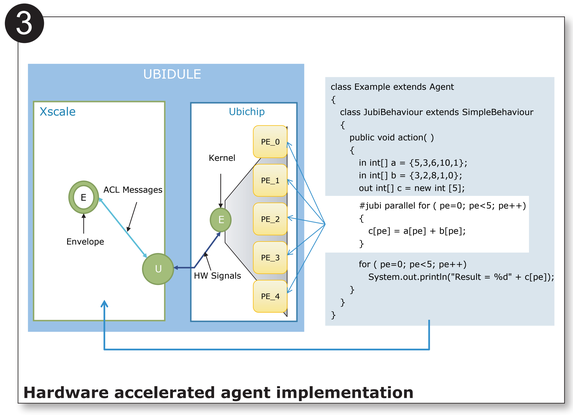
Figure 2 shows how agents can be implemented either in software – in Java for execution on the XScale –, in hardware – using the Ubichip assembler – or using a "software restricted" method, that is compiled into an Ubichip-executable form using a specific compiler. In the last case, the JubiCompiler tools disembody the agent to perform some of the computations transparently on the Ubichip (see Figure 3), providing a unique tool to facilitate the analysis of parallelism and synchronization on the application performance and output.

2



Perplexus modeling framework

3



Hardware accelerated agent implementation

DISCUSSION

This poster presents the novel implementation of a neural system simulator on an hybrid multi-agent hardware platform within the Perplexus project. The proposed architecture provides a scalable and flexible framework for biologically plausible neural network modeling.

Future work is aimed at realizing the complete hardware acceleration features. The intended implementation of the framework on mobile robots will provide a testable platform for the phylogenetic processes.

ACKNOWLEDGMENTS

This work is supported by the European Commission Future and Emerging Technologies Complex Systems program, grant #034632 (PERPLEXUS).

REFERENCES

http://www.perplexus.org/

[1] O. Brousse, G. Sassatelli, T. Gil, M. Robert, L. Torres, E. Sanchez, J.M. Moreno, A. Villa, H. Volken, A. Napieralski, F. Mondada (2008). *Perplexus Project Programming Framework*, First International Conference on Embedded Systems & Critical Applications, ICESCA'08

[2] J. Iglesias, O.K. Chibirova, A.E.P. Villa (2007). *Nonlinear Dynamics Emerging in Large Scale Neural Networks with Ontogenetic and Epigenetic Processes*, Lecture Notes in Computer Sciences 4668, 579-588