

[Home](#)

Intelligent Autonomous Systems with Build-in Digital Spiking Neural Networks

Introduction

This is the main web page of the *Intelligent Autonomous Systems with Build-in Digital Spiking Neural Networks* research project which is funded by the [FWO](#) (national research fund). It is a collaboration between the [PARIS](#) research group of the UGhent and the [PMA](#) research group of the K.U.Leuven. It is a four year spanning project which started january 1 2005.

The main research objective of this project is to set up fundamental research to the possibilities and the applicability of hardware implementations of spiking neural models for supporting self-learning properties in robotics.

Official Project Proposal

Aim

Research on intelligent, autonomous systems is in the centre of interest internationally and the possible applications appeal to one's imagination. In the field of robotics, e.g., autonomy is an increasingly desired feature. The robot should be sufficiently intelligent to perform a number of tasks autonomously, without the need to explicitly program these tasks beforehand. An example is finding a path in a building that is collapsing or burning, or understanding not clearly specified wishes of a user, based on person-related intention estimation (e.g., for a handicapped person operating a wheelchair). A second aspect gaining importance in robotics is team work, the execution of a task by many small robots instead of one big one. Recent projects have investigated how such robots can collaborate and can learn how to react to each other. These two evolutions in robotics (autonomy and team work) imply that the robot(s) is (are) able to learn. A much used method is reinforcement learning in which a good solution is rewarded and a bad one is reproved. Another method is learning by imitation, in which the desired behaviour is learned by showing an example. The research group PMA has realised on-line learning behaviour for assembly and manipulation applications.

For the exploitation of self-learning abilities, techniques using neural networks are very appropriate. Neural networks are inspired by the operation of the brain and are particularly suited for learning tasks. Neural networks also allow a system to react well to new events even though it was not especially trained for that task. This concept is called generalisation. Additionally, neural networks can be trained during their operation to allow them to handle unexpected situations. Researchers at NASA have shown that neural networks are very well suited (better than classical techniques) to react to unexpected events in airplanes (such as

broken control parts and motors, &). They even plan to equip all airplanes with such neural control units from 2010 onwards [A99].

Yet, neural networks have never brought a breakthrough for intelligent autonomous systems, a breakthrough one had expected from them for decades. One reason for this is that current processor-based implementations of neural networks are too slow and computation-intensive to easily integrate them in various applications. Therefore, the most obvious solution is a hardware implementation, which is much faster and more compact. A hardware implementation can also be inherently parallel, allowing the computations in many network nodes to be done at the same time and therefore more efficiently. But designing an ASIC (Application Specific Integrated Circuit) is very costly, especially if only a limited number of chips are needed. ASICs also allow too little flexibility to change the neural network in a later stage. The introduction of FPGAs (Field Programmable Gate Arrays) in the 80s has made a hardware solution much cheaper, especially for smaller quantities. They also enable the possibility to reprogram the chips at a later time and thus stimulate the optimisation of the neural network implementation. However, a hardware implementation of neural networks is hampered by the fact that today's neural network models are based on the processor computation model. Hence they are less well-suited to be implemented in hardware. Therefore, hardware neural networks have not taken a high flight either.

Of course, also FPGAs evolve with the strong miniaturisation. Current FPGAs therefore are powerful and relatively fast components. Modern FPGAs contain simple processors that are integrated on the FPGA chip. These could be used to implement classical neural networks and to exploit the FPGA's reprogrammability. But still then the FPGA is not used efficiently and the drawbacks of a processor-based neural network remain.

Recently, new neural models have been investigated that are much more closely coupled with the binary computations of digital computation and that are even more tightly linked to the brain's operation than classical neural models [MB99]. These spiking neural models only use bits for the communication between neurons. The neurons themselves are simply weighted leaking integrators that can fire and that can be reset as soon as a threshold value is reached. Because the communication between network nodes now is handled by spikes instead of by floating point values, we should explore ways of representing the mostly analogue information from the surroundings into spikes. Several coding techniques are possible. Recently, Floriano et al. showed that hardware spiking neural networks are quite useful for autonomous robotics [FSCB02] and that they are actually better suited for the task than classical neural networks. However, very simple neuron models are used, as well as a global training algorithm (a genetic algorithm). This is mostly unwanted. The research area of spiking neural networks is still in full development, especially in Europe.

An additional benefit of spiking neural networks is the fact that they are proven to be more powerful than the classical neural networks [M97]. This makes the corresponding hardware implementation not only more efficient, but also theoretically more powerful than a hardware implementation of a classical neural model.

By combining spiking neural models with a hardware implementation on flexible FPGA architectures, neural networks are becoming a viable solution that opens the door for a plethora of possibilities for intelligent autonomous systems, such as self-learning robots,

robot-assisted surgery, distributed modular control, service robots for elderly and handicapped people, user-dependent changes in ambient intelligence, immobots, etc.

This project is based primarily on the joining of know-how and experience of the research groups PARIS (UGent) and PMA (K.U.Leuven). The research group PARIS especially contributes its experience on the young area of spiking neural network models. The research group can also rely on its experience with and study and even design of FPGA components. PARIS wants to focus the broadly performed research at (still fundamental) research that is applicable in the important area of the autonomous intelligent systems. In that domain, the research group PMA is a pioneer, more in particular in robotics, mechatronics holonic production systems and decentralised and modular control of drive systems. Recent research work that will be consolidated in this project contains learning robots, the realisation of reinforcement learning in assembly, intelligent wheelchair control and ambient intelligence. The combination of these two very different research domains potentially will deliver very interesting research results. This is because robotics, and the work done by PMA in particular, seems to be the ideal application domain for the use of hardware implementations of the spiking neural models PARIS works on. It is clear the interdisciplinary cooperation in this is required.

Objectives

The main objective of this project is to set up fundamental basic research to the possibilities and the applicability of hardware implementations of spiking neural models for supporting self-learning properties in robotics.

This objective contains two intertwined research parts. On the one hand, spiking neural network models are a very young research area and it still needs a lot of fundamental research to optimise and extend the model itself. On the other hand, the right techniques have to be found to apply reinforcement learning or backpropagation on classical robot problems. Here, we want to test the evolution in spiking neural models against its application in robotics. Vice versa, we will investigate the possibilities of using spiking neural models for robotics. The synergy between the two research lines therefore is strongly present in this project and actually is an integral part of it.

In the area of extensions to the spiking neural model, we have three main objectives:

- Further research of the information coding in spikes.

- Setting up training algorithms for spiking neural networks, both backpropagation and reinforcement learning.

- Searching for the right spiking neural model aiming at an implementation in reconfigurable hardware.

In the application area of robotics, we aim at theoretical research on the use of reinforcement learning and backpropagation using spiking neural networks. Applications in the long term are robot manipulation serving elderly or handicapped people, person-driven adaptation for wheelchair control, or collaborating robots. We will also study the neural network topology needed for self-learning features and we will investigate the models that can or should be integrated.

Design and methodology

The research will be performed by two research groups in close synergy. The research group PARIS (UGent) will focus on the properties and working of spiking neural models itself; The research group PMA (K.U.Leuven) will particularly study the application area and treat the requirements for it. In detail, the research effort will consist of the following tasks in order to realise the objectives.

1. Existing coding techniques will be investigated on their usefulness for a hardware implementation of spiking neural models and for realising learning behaviour in intelligent autonomous systems. In this project task, we study the coding precision and implementation simplicity, a few encoders/decoders for the various coding techniques and the possible impact of the coding on the power of the neural network. We also investigate which codes are best suited for the typical problems in intelligent autonomous systems.
2. Based on the results of part 1, we will try to suggest new coding techniques that combine the best features of various existing coding techniques or that add new features, missing from the existing techniques. Of course we will also have to investigate the benefits and drawbacks of these techniques. E.g., in the neuro-literature, a recent paper notes that decoding with two filters improves the reconstruction quality.
3. Today, the number of learning rules for spiking neural networks is rather limited: there are local rules that allow non-supervised learning, there exist global learning rules that are not applicable for on-line learning and there are backpropagation rules, but only for time coding. We will further investigate these learning rules and extend them to ensure they are useful in hardware implementations and for on-line learning of the neural networks. We will extend the existing backpropagation rule such that other network parameters can also be trained and so that the network topology is less restricted. We will also search for training algorithms for other codings than time coding to compare the power of neural networks with each of the codings. In particular, we will search new learning rules (or extensions of existing ones) to realise reinforcement learning and backpropagation, in combination with a certain coding technique. We also will have to research the robustness of various algorithms.
4. In order to get a first impression on the problems that occur with an effective hardware implementation, we will research possible implementations of a single spiking neuron in reconfigurable hardware. Issues such as computing power in relation to silicon area and length of time will be investigated.
5. Next, we investigate ways to combine neurons to form a network. One of the main features of such a network is its topology. The choice of topology will largely depend on the previous choices of coding techniques and learning rules. However, the chosen application area will have a big impact too. Another issue here is how to map a high dimensional topology on two-dimensional hardware. In this task, we will investigate what network topology is most suited for autonomous learning systems, in combination with the previously found most appropriate coding techniques and learning rules. We will also study how the FPGA-type reconfigurable interconnects could be used to project the high dimensional topology onto a two-dimensional topology.
6. We also have to search a good way to simulate the neural networks. This is the first step towards a hardware implementation of the entire network (whose topology is fixed in task

- 5). Probably, the network will not be entirely implemented in parallel on the FPGA but it might be partly multiplexed in time. To obtain a well thought-out network partitioning, simulations are required.
7. In the last step, we will build the various neurons into a single hardware implementation of the entire network. Conclusions from previous steps will be combined here into a hardware implementation of a spiking neural network. Furthermore, we will investigate the training of such hardware neural networks. Should the training infrastructure be put in the hardware itself, or is it possible that a peripheral processor takes this task and that this constitutes an efficient whole?
 8. During and after the hardware implementation of the entire neural network, we want to consider the applicability and possible impact of spiking neural networks for intelligent autonomous systems. An important element here is the way in which the neural network interacts with the world. Possible challenges are, e.g.
 - realisation of local intelligence and learning behaviour in a decentralised modular system;
 - realisation of basic skills and learning behaviour within behaviour-based autonomous systems.

With this research we want to gain new insights in these topics.

In each of these project tasks the collaboration between both research groups is essential because both hardware aspects as well as aspects of self-learning systems are involved. The research groups therefore will hold frequent technical meetings in which the progress of one group is put to the test against the know-how and research results of the other group.

References

[A99]

Annon. Intelligent Flight Control: Advanced Concept Program Final Report. The Boeing Company, BOEING-STL 99P0040, 1999.

[FSCB02]

D. Floreano, N. Schoeni, G. Caprari en J. Blynel. Evolutionary Bits n Spikes. In Russell K. Standish, Mark A. Bedau and Hussein A. Abbass, editors, Artificial Life VIII: Proceedings, page 335. The MIT Press, 2002.

[MB99]

W. Maass en C. M. Bishop. Pulsed Neural Networks. MIT Press, Cambridge, MA, 1999.

[M97]

W. Maass. Noisy spiking neurons with temporal coding have more computational power than sigmoidal neurons. Michael C. Mozer, Micael I. Jordan, and Thomas Persche, editors, Advances in Neural Information Processing Systems, volume 9, page 211. The MIT Press, 1997.