The CAM-Brain Machine (CBM): An FPGA Based Tool for Evolving a 75 Million Neuron Artificial Brain to Control a Lifesized Kitten Robot

HUGO DE GARIS
STARLAB, Rue Engeland 555, B-1180, Brussels, Belgium
degaris@starlab.net, http://foobar.starlab.net/~degaris

MICHAEL KORKIN AND GARY FEHR
GENOBYTE, Inc., 1200 Pearl Street, Suite 65, Boulder CO 80302
korkin,fehr@genobyte.com, http://www.genobyte.com

Abstract. This article introduces the “CAM-Brain Machine” (CBM), an FPGA based piece of hardware which implements a genetic algorithm (GA) to evolve a cellular automata (CA) based neural network circuit module, of approximately 1,000 neurons, in about a second, i.e., a complete run of a GA, with 10,000s of circuit growths and performance evaluations. Up to 65,000 of these modules, each of which is evolved with a humanly specified function, can be downloaded into a large RAM space, and interconnected according to humanly specified artificial brain architectures. This RAM, containing an artificial brain with up to 75 million neurons, is then updated by the CBM at a rate of 130 billion CA cells per second. Such speeds should enable real time control of robots and hopefully the birth of a new research field that we call “brain building.” The first such artificial brain, to be built in 2000 and beyond, will be used to control the behaviors of a life sized robot kitten called “Robokitty.”

Keywords: evolvable hardware, artificial brain, neural networks, genetic algorithm, kitten robot

1. Introduction

This article introduces the “CAM-Brain Machine” (CBM) (Korkin et al., 1997), a Xilinx XC6264 FPGA (Xilinx, Inc., 1996) based piece of hardware that is used to evolve 3D cellular automata based neural network (Rumelhart and McClelland, 1986) circuit modules at electronic speeds, that is in about a second per module. 65,000 of these modules can then be assembled into a large RAM space according to humanly specified artificial brain architectures. This RAM is updated by the CBM fast enough (130 billion CA cell updates/sec) for real time control of robots. Four of these CBMs have already been or are about to be delivered (the first to de Garis’s previous lab, ATR in Kyoto, Japan, the second to Belgium’s Lernout and Hauspie speech processing company, the third is to be delivered to STARLAB in June 2000, and GENOBYTE will have its own also in June 2000).

The CBM is the essential tool in the “Artificial Brain (CAM-Brain) Project” (de Garis, 1994; de Garis et al., 1998; website http://Foobar.starlab.net/~degaris/journal.html), which at the time of writing (Spring 2000), has been running for 7+ years. Although the focus of this article is on the functional principles and design of the CBM, a certain background needs to be provided so that the motivation for its construction is understood.

The basic (and rather ambitious) aim of the CAM-Brain Project as first stated in 1993 was to build an artificial brain containing a billion artificial neurons by the year 2001. The actual figure in 2000 is maximum 75 million, but the billion figure is still reachable if we really want. The Brain Builder teams at STARLAB and GENOBYTE are hoping that the CBM will revolutionize the field of neural networks (by creating neural systems with tens of millions of artificial neurons, rather than just the conventional tens to
hundreds), and will create a new research field called “Brain Building.” The CBM will make practical the creation of artificial brains, which are defined to be assemblages of tens of thousands (and higher magnitudes) of evolved neural net modules into humanly defined artificial brain architectures. An artificial brain will consist of a large RAM memory space, into which individual CA modules are downloaded once they have been evolved. The CA cells in this RAM will be updated by the CBM fast enough for real time control of a robot kitten “Robokitty.”

Since the neural net model used to fit into state-of-the-art evolvable electronics has to be simple, the signaling states of the neural net were chosen to be 1 bit binary. We label this model “CoDi-1Bit” (Gers, et al., 1997) (CoDi = Collect & Distribute). This article will summarize the principles of this 1 bit neural signaling model, since the CBM is an electronic implementation of it. We realize that limiting ourselves to only 1 bit per neural signal (to fit into the Xilinx XC6264 chips), is rather severe (although nature uses a 1 bit signal scheme with its evoked potentials, i.e., the spikes in the axons), so it is possible that future versions of the CBM may use multibit neural signaling to obtain higher “evolvability” of neural module functionality.

The remainder of this article is structured as follows. Section 2 briefly covers three topics already published elsewhere (e.g., de Garis, website; de Garis et al., 1998). The first is the “CoDi-1Bit” neural net model that is implemented by the CAM-Brain Machine (CBM). The second is the representation that our team chose to interpret the 1 bit signals which are input to and output from the CoDi modules (we call this representation “SIIC” = Spike Interval Information Coding). This representation is important because the CBM measures the “fitness” (i.e., the performance measure of the evolving circuit) using analog output values obtained by convoluting the binary outputs of the module with a digitized convolution function. The third shows how analog time-dependent signals can be converted into spike trains (bit strings of 0s and 1s) to be input into CoDi modules using the so-called “HSA” (Hough Spiker Algorithm). The SIIC (spiketrain to analog signal conversion) and the HSA (analog signal to spiketrain conversion) allow users (evolutionary engineers) to think entirely in analog terms when specifying input signals and target (desired) output signals, which is much easier than thinking in terms of spike intervals (the number of 0s between the 1s). This analog thinking for evolutionary engineers simplifies the evolution of modules, and overcomes the limitation to some extent of the 1 bit binary signaling of the CoDi modules (and hence the CBM). Section 3, the heart of this article, provides a detailed summary of CBM design and functionality, using the ideas already discussed briefly in the earlier sections. Since an artificial brain without a body (such as a robot) seems rather pointless, Section 4 introduces early work on the behavioral repertoire and mechanical design of the kitten robot “Robokitty” that our artificial brain will control. Section 5 talks about some related work, and Section 6 concludes.

2. The CoDi-1Bit Neural Network Model and Digital/Analog Conversion

The CBM implements the so called “CoDi” (i.e., Collect and Distribute) (Gers et al., 1997) cellular automata based neural network model. It is a simplified form of an earlier model developed at ATR (Kyoto, Japan) in the summer of 1996, with two goals in mind. One was to make neural network functioning much simpler and more compact compared to the original ATR model, so as to achieve considerably faster evolution runs on the CAM-8 (Cellular Automata Machine), a dedicated hardware tool developed at Massachusetts Institute of Technology in 1989.

In order to evolve one neural module, a population of 30–100 modules is run through a genetic algorithm (Goldberg, 1989) for 200–600 generations, resulting in up to 60,000 different module evaluations. Each module evaluation consists of—firstly, growing a new set of axonic and dendritic trees, guided by the module’s chromosome (which provide the growth instructions for the trees). These trees interconnect several hundred neurons in the 3D cellular automata space of 13,824 cells ($24 \times 24 \times 24$). Evaluation is continued by sending spiketrains to the module through its efferent axons (external connections) to evaluate its performance (fitness) by looking at the outgoing spiketrains. This typically requires up to 1000 update cycles for all the cells in the module.

On the MIT CAM-8 machine, it takes up to 69 minutes to go through 829 billion cell updates needed to evolve a single neural module, as described above. A simple “insect-like” artificial brain may have hundreds of thousands of neurons arranged into ten thousand modules. It would take 500 days (running 24 hours a day) to finish the computations.

Another limitation was apparent in the full brain simulation mode, involving thousands of modules,