Real-Time Control of a Tokamak Plasma Using Neural Networks

Chris M Bishop
Neural Computing Research Group
Department of Computer Science
Aston University
Birmingham, B4 7ET, U.K.
c.m.bishop@aston.ac.uk

**Paul S Haynes, Mike E U Smith, Tom N Todd,
David L Trotman and Colin G Windsor**
AEA Technology, Culham Laboratory,
Oxfordshire OX14 3DB
(Euratom/UKAEA Fusion Association)

Abstract

This paper presents results from the first use of neural networks for the real-time feedback control of high temperature plasmas in a tokamak fusion experiment. The tokamak is currently the principal experimental device for research into the magnetic confinement approach to controlled fusion. In the tokamak, hydrogen plasmas, at temperatures of up to 100 Million K, are confined by strong magnetic fields. Accurate control of the position and shape of the plasma boundary requires real-time feedback control of the magnetic field structure on a time-scale of a few tens of microseconds. Software simulations have demonstrated that a neural network approach can give significantly better performance than the linear technique currently used on most tokamak experiments. The practical application of the neural network approach requires high-speed hardware, for which a fully parallel implementation of the multilayer perceptron, using a hybrid of digital and analogue technology, has been developed.

1   INTRODUCTION

Fusion of the nuclei of hydrogen provides the energy source which powers the sun. It also offers the possibility of a practically limitless terrestrial source of energy. However, the harnessing of this power has proved to be a highly challenging problem. One of the most promising approaches is based on magnetic confinement of a high temperature ($10^7 - 10^8$ Kelvin) plasma in a device called a tokamak (from the Russian for 'toroidal magnetic chamber') as illustrated schematically in Figure 1. At these temperatures the highly ionized plasma is an excellent electrical conductor, and can be confined and shaped by strong magnetic fields. Early tokamaks had plasmas with circular cross-sections, for which feedback control of the plasma position and shape is relatively straightforward. However, recent tokamaks, such as the COMPASS experiment at Culham Laboratory, as well as most next-generation tokamaks, are designed to produce plasmas whose cross-sections are strongly non-circular. Figure 2 illustrates some of the plasma shapes which COMPASS is designed to explore. These novel cross-sections provide substantially improved energy confinement properties and thereby significantly enhance the performance of the tokamak.
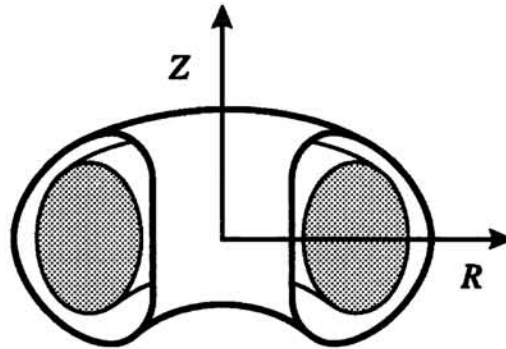

Figure 1: Schematic cross-section of a tokamak experiment showing the toroidal vacuum vessel (outer D-shaped curve) and plasma (shown shaded). Also shown are the radial (R) and vertical (Z) coordinates. To a good approximation, the tokamak can be regarded as axisymmetric about the Z-axis, and so the plasma boundary can be described by its cross-sectional shape at one particular toroidal location.

Unlike circular cross-section plasmas, highly non-circular shapes are more difficult to produce and to control accurately, since currents through several control coils must be adjusted simultaneously. Furthermore, during a typical plasma pulse, the shape must evolve, usually from some initial near-circular shape. Due to uncertainties in the current and pressure distributions within the plasma, the desired accuracy for plasma control can only be achieved by making real-time measurements of the position and shape of the boundary, and using error feedback to adjust the currents in the control coils.

The physics of the plasma equilibrium is determined by force balance between the

<div align="center">

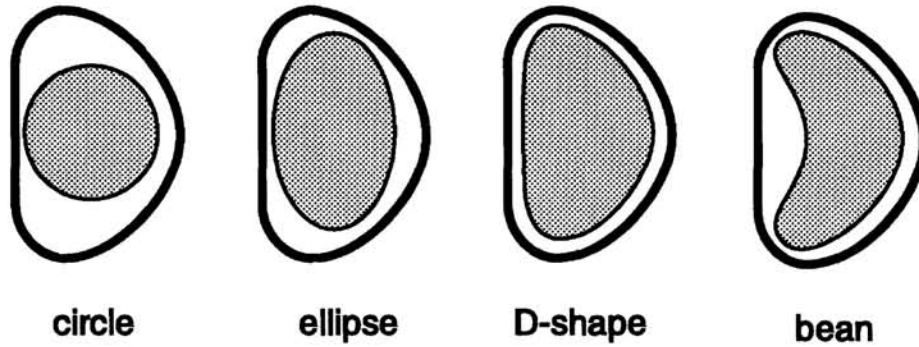**circle**   **ellipse**   **D-shape**   **bean**

</div>

Figure 2: Cross-sections of the COMPASS vacuum vessel showing some examples of potential plasma shapes. The solid curve is the boundary of the vacuum vessel, and the plasma is shown by the shaded regions.

thermal pressure of the plasma and the pressure of the magnetic field, and is relatively well understood. Particular plasma configurations are described in terms of solutions of a non-linear partial differential equation called the Grad-Shafranov (GS) equation. Due to the non-linear nature of this equation, a general analytic solution is not possible. However, the GS equation can be solved by iterative numerical methods, with boundary conditions determined by currents flowing in the external control coils which surround the vacuum vessel. On the tokamak itself it is changes in these currents which are used to alter the position and cross-sectional shape of the plasma. Numerical solution of the GS equation represents the standard technique for post-shot analysis of the plasma, and is also the method used to generate the training dataset for the neural network, as described in the next section. However, this approach is computationally very intensive and is therefore unsuitable for feedback control purposes.

For real-time control it is necessary to have a fast (typically $\leq 50\mu sec.$) determination of the plasma boundary shape. This information can be extracted from a variety of diagnostic systems, the most important being local magnetic measurements taken at a number of points around the perimeter of the vacuum vessel. Most tokamaks have several tens or hundreds of small pick up coils located at carefully optimized points around the torus for this purpose. We shall represent these magnetic signals collectively as a vector m.

For a large class of equilibria, the plasma boundary can be reasonably well represented in terms of a simple parameterization, governed by an angle-like variable θ, given by

$$
\begin{aligned}
R(\theta) &= R_0 + a\cos(\theta + \delta\sin\theta) \\
Z(\theta) &= Z_0 + a\kappa\sin\theta
\end{aligned}
\tag{1}
$$

where we have defined the following parameters

R_0 radial distance of the plasma center from the major axis of the torus,
Z_0 vertical distance of the plasma center from the torus midplane,
a minor radius measured in the plane $Z = Z_0$,
κ elongation,
δ triangularity.

We denote these parameters collectively by y_k. The basic problem which has to be addressed, therefore, is to find a representation for the (non-linear) mapping from the magnetic signals m to the values of the geometrical parameters y_k, which can be implemented in suitable hardware for real-time control.

The conventional approach presently in use on many tokamaks involves approximating the mapping between the measured magnetic signals and the geometrical parameters by a single linear transformation. However, the intrinsic non-linearity of the mappings suggests that a representation in terms of feedforward neural networks should give significantly improved results (Lister and Schnurrenberger, 1991; Bishop *et al.*, 1992; Lagin *et al.*, 1993). Figure 3 shows a block diagram of the control loop for the neural network approach to tokamak equilibrium control.
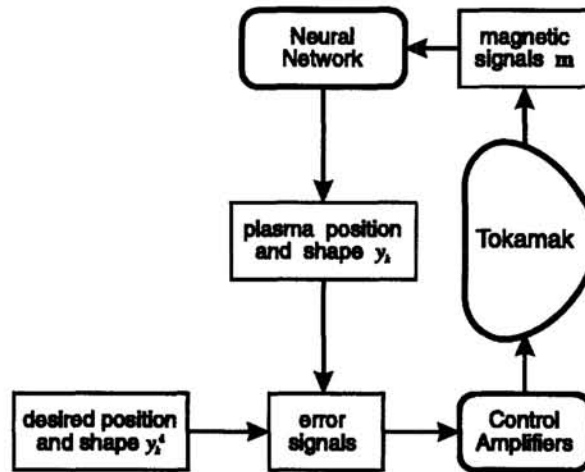

Figure 3: Block diagram of the control loop used for real-time feedback control of plasma position and shape.

2 SOFTWARE SIMULATION RESULTS

The dataset for training and testing the network was generated by numerical solution of the GS equation using a free-boundary equilibrium code. The data base currently consists of over 2,000 equilibria spanning the wide range of plasma positions and shapes available in COMPASS. Each equilibrium configuration takes several minutes to generate on a fast workstation. The boundary of each configuration is then fitted using the form in equation 1, so that the equilibria are labelled with the appropriate values of the shape parameters. Of the 120 magnetic signals available on COMPASS which could be used to provide inputs to the network, a

subset of 16 has been chosen using sequential forward selection based on a linear representation for the mapping (discussed below).

It is important to note that the transformation from magnetic signals to flux surface parameters involves an exact linear invariance. This follows from the fact that, if all of the currents are scaled by a constant factor, then the magnetic fields will be scaled by this factor, and the geometry of the plasma boundary will be unchanged. It is important to take advantage of this prior knowledge and to build it into the network structure, rather than force the network to learn it by example. We therefore normalize the vector m of input signals to the network by dividing by a quantity proportional to the total plasma current. Note that this normalization has to be incorporated into the hardware implementation of the network, as will be discussed in Section 3.
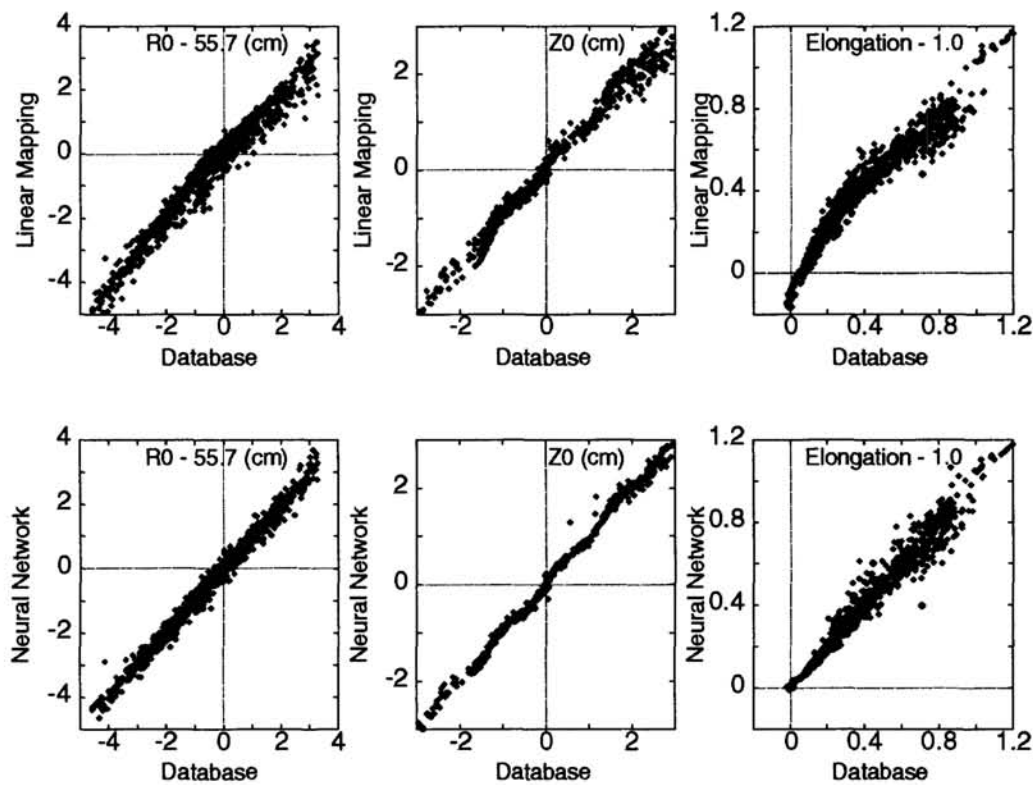

Figure 4: Plots of the values from the test set versus the values predicted by the linear mapping for the 3 equilibrium parameters, together with the corresponding plots for a neural network with 4 hidden units.

The results presented in this paper are based on a multilayer perceptron architecture having a single layer of hidden units with 'tanh' activation functions, and linear output units. Networks are trained by minimization of a sum-of-squares error using a standard conjugate gradients optimization algorithm, and the number of hidden

units is optimized by measuring performance with respect to an independent test set. Results from the neural network mapping are compared with those from the optimal linear mapping, that is the single linear transformation which minimizes the same sum-of-squares error as is used in the neural network training algorithm, as this represents the method currently used on a number of present day tokamaks.

Initial results were obtained on networks having 3 output units, corresponding to the values of vertical position Z_0, major radius R_0, and elongation κ; these being parameters which are of interest for real-time feedback control. The smallest normalized test set error of 11.7 is obtained from the network having 16 hidden units. By comparison, the optimal linear mapping gave a normalized test set error of 18.3. This represents a reduction in error of about 30% in going from the linear mapping to the neural network. Such an improvement, in the context of this application, is very significant.

For the experiments on real-time feedback control described in Section 4 the currently available hardware only permitted networks having 4 hidden units, and so we consider the results from this network in more detail. Figure 4 shows plots of the network predictions for various parameters versus the corresponding values from the test set portion of the database. Analogous plots for the optimal linear map predictions versus the database values are also shown. Comparison of the corresponding figures shows the improved predictive capability of the neural network, even for this sub-optimal network topology.

3   HARDWARE IMPLEMENTATION

The hardware implementation of the neural network must have a bandwidth of $\geq$ 20 kHz in order to cope with the fast timescales of the plasma evolution. It must also have an output precision of at least (the the analogue equivalent of) 8 bits in order to ensure that the final accuracy which is attainable will not be limited by the hardware system. We have chosen to develop a fully parallel custom implementation of the multilayer perceptron, based on analogue signal paths with digitally stored synaptic weights (Bishop *et al.*, 1993). A VME-based modular construction has been chosen as this allows flexibility in changing the network architecture, ease of loading network weights, and simplicity of data acquisition. Three separate types of card have been developed as follows:

- Combined 16-input buffer and signal normalizer.

  This provides an analogue hardware implementation of the input normalization described earlier.

- 16 × 4 matrix multiplier

  The synaptic weights are produced using 12 bit frequency-compensated multiplying DACs (digital to analogue converters) which can be configured to allow 4-quadrant multiplication of analogue signals by a digitally stored number.

- 4-channel sigmoid module

  There are many ways to produce a sigmoidal non-linearity, and we have opted for a solution using two transistors configured as a long-tailed-pair,

to generate a 'tanh' sigmoidal transfer characteristic. The principal drawback of such an approach is the strong temperature sensitivity due to the appearance of temperature in the denominator of the exponential transistor transfer characteristic. An elegant solution to this problem has been found by exploiting a chip containing 5 transistors in close thermal contact. Two of the transistors form the long-tailed pair, one of the transistors is used as a heat source, and the remaining two transistors are used to measure temperature. External circuitry provides active thermal feedback control, and stability to changes in ambient temperature over the range 0°C to 50°C is found to be well within the acceptable range.

The complete network is constructed by mounting the appropriate combination of cards in a VME rack and configuring the network topology using front panel interconnections. The system includes extensive diagnostics, allowing voltages at all key points within the network to be monitored as a function of time via a series of multiplexed output channels.

4   RESULTS FROM REAL-TIME FEEDBACK CONTROL

Figure 5 shows the first results obtained from real-time control of the plasma in the COMPASS tokamak using neural networks. The evolution of the plasma elongation, under the control of the neural network, is plotted as a function of time during a plasma pulse. Here the desired elongation has been preprogrammed to follow a series of steps as a function of time. The remaining 2 network outputs (radial position R_0 and vertical position Z_0) were digitized for post-shot diagnosis, but were not used for real-time control. The solid curve shows the value of elongation given by the corresponding network output, and the dashed curve shows the post-shot reconstruction of the elongation obtained from a simple 'filament' code, which gives relatively rapid post-shot plasma shape reconstruction but with limited accuracy. The circles denote the elongation values given by the much more accurate reconstructions obtained from the full equilibrium code. The graph clearly shows the network generating the required elongation signal in close agreement with the reconstructed values. The typical residual error is of order 0.07 on elongation values up to around 1.5. Part of this error is attributable to residual offset in the integrators used to extract magnetic field information from the pick-up coils, and this is currently being corrected through modifications to the integrator design. An additional contribution to the error arises from the restricted number of hidden units available with the initial hardware configuration. While these results represent the first obtained using closed loop control, it is clear from earlier software modelling of larger network architectures (such as 32–16–4) that residual errors of order a few % should be attainable. The implementation of such larger networks is being persued, following the successes with the smaller system.
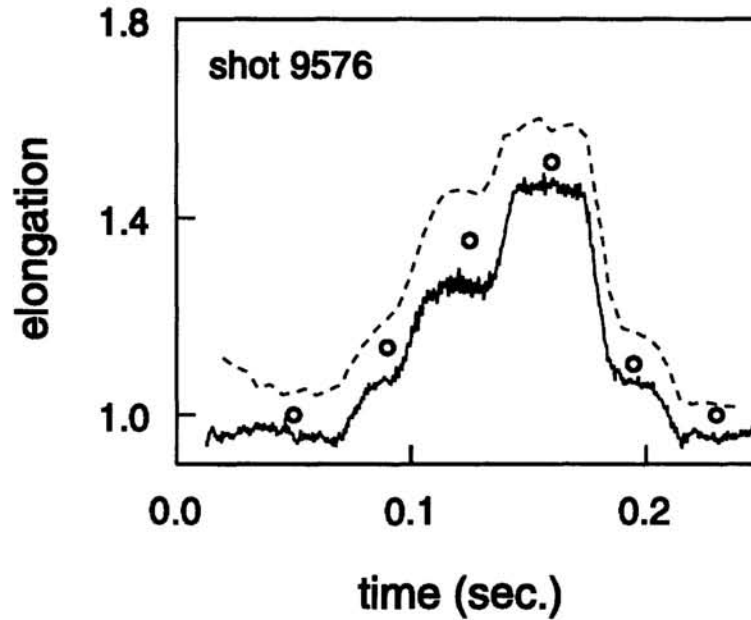
Acknowledgements

Figure 5: Plot of the plasma elongation κ as a function of time during shot no. 9576 on the COMPASS tokamak, during which the elongation was being controlled in real-time by the neural network.

References

Bishop C M, Cox P, Haynes P S, Roach C M, Smith M E U, Todd T N and Trotman D L, 1992. A neural network approach to tokamak equilibrium control. In *Neural Network Applications*, Ed. J G Taylor, Springer Verlag, 114–128.

Bishop C M, Haynes P S, Roach C M, Smith M E U, Todd T N, and Trotman D L. 1993. Hardware implementation of a neural network for plasma position control in COMPASS-D. In *Proceedings of the 17th. Symposium on Fusion Technology*, Rome, Italy. **2** 997–1001.

Lagin L, Bell R, Davis S, Eck T, Jardin S, Kessel C, Mcenerney J, Okabayashi M, Popyack J and Sauthoff N. 1993. Application of neural networks for real-time calculations of plasma equilibrium parameters for PBX-M, In *Proceedings of the 17th. Symposium on Fusion Technology*, Rome, Italy. **2** 1057–1061.

Lister J B and Schnurrenberger H. 1991. Fast non-linear extraction of plasma parameters using a neural network mapping. *Nuclear Fusion*. **31**, 1291–1300.

Pulsestream Synapses with Non-Volatile Analogue Amorphous-Silicon Memories.

A.J. Holmes, A.F. Murray, S. Churcher and J. Hajto
Department of Electrical Engineering
University of Edinburgh
Edinburgh, EH9 3JL

M. J. Rose
Dept. of Applied Physics and Electronics,
Dundee University
Dundee DD1 4HN

Abstract

A novel two-terminal device, consisting of a thin 1000Å layer of p^+ a-Si:H sandwiched between Vanadium and Chromium electrodes, exhibits a non-volatile, analogue memory action. This device stores synaptic weights in an ANN chip, replacing the capacitor previously used for dynamic weight storage. Two different synapse designs are discussed and results are presented.

1   INTRODUCTION

Analogue hardware implementations of neural networks have hitherto been hampered by the lack of a straightforward (local) analogue memory capability. The ideal storage mechanism would be compact, non-volatile, easily reprogrammable, and would not interfere with the normal silicon chip fabrication process.

Techniques which have been used to date include resistors (these are not generally reprogrammable, and suffer from being large and difficult to fabricate with any accuracy), dynamic capacitive storage [4] (this is compact, reprogrammable and simple, but implies an increase in system complexity, arising from off-chip refresh circuitry),

EEPROM ("floating gate") memory [5] (which is compact, reprogrammable, and non-volatile, but is slow, and cannot be reprogrammed in situ), and local digital storage (which is non-volatile, easily programmable and simple, but consumes area horribly).

Amorphous silicon has been used for synaptic weight storage [1, 2], but only as either a high-resistance fixed weight medium or a binary memory.

In this paper, we demonstrate that novel amorphous silicon memory devices can be incorporated into standard CMOS synapse circuits, to provide an analogue weight storage mechanism which is compact, non-volatile, easily reprogrammable, and simple to implement.

2   a-Si:H MEMORY DEVICES

The a-Si:H analogue memory device [3] comprises a 1000Å thick layer of amorphous silicon (p^+ a-Si:H) sandwiched between Vanadium and Chromium electrodes.

The a-Si device takes the form of a two-terminal, programmable resistor. It is an "add-on" to a conventional CMOS process, and does not demand that the normal CMOS fabrication cycle be disrupted. The a-Si device sits on top of the completed chip circuitry, making contact with the CMOS arithmetic elements via holes cut in the protective passivation layer, as shown in Figure 1.
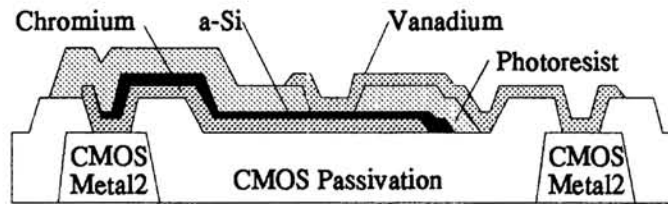

Figure 1: The construction of a-Si:H Devices on a CMOS chip

After fabrication a number of electronic procedures must be performed in order to program the device to a given resistance state.

Programming, and Pre-Programming Procedures

Before the a-Si device is usable, the following steps must be carried out:

- Forming: This is a once-only process, applied to the a-Si device in its "virgin" state, where it has a resistance of several MΩ. A series of 300ns pulses, increasing in amplitude from 5v to 14v, is applied to the device electrodes. This creates a vertical conducting channel or filament whose approximate resistance is 1KΩ. This filament can then be programmed to a value in the range 1KΩ to 1MΩ. The details of the physical mechanisms are not yet fully established, but it is clear that conduction occurs through a narrow (sub-micron) conducting channel.

- Write: To decrease the device's resistance, negative "Write", pulses are applied.

- Erase: To increase the device's resistance, positive "Erase", pulses are applied.

- Usage: Pulses below 0.5v do not change the device resistance. The resistance can therefore be utilised as a weight storage medium using a voltage of less than 0.5v without causing reprogramming.

Programming pulses, which range between 2v and 5v, are typically 120ns in duration. Programming is therefore much faster than for other EEPROM (floating gate) devices used in the same context, which use a series of $100\mu s$ pulses to set the threshold voltage [5].

The following sections describe synapse circuits using the a-Si:H devices. These synapses use the reprogrammable a-Si:H resistor in the place of a storage capacitor or EEPROM cell. These new synapses were implemented on a chip referred to as ASiTEST2, consisting of five main test blocks, each comprising of four synapses connected to a single neuron.

3   The EPSILON based synapse

The first synapse to be designed used the a-Si:H resistor as a direct replacement for the storage capacitor used in the EPSILON [4] synapse.
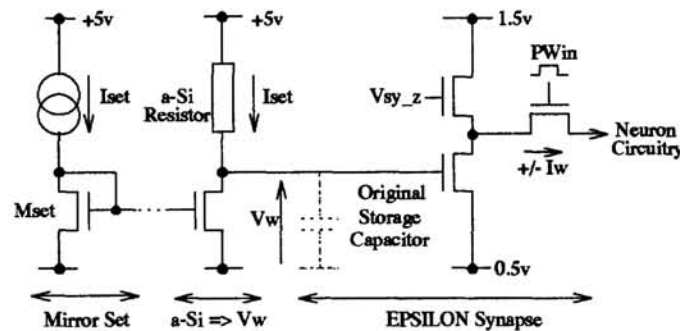

Figure 2: The EPSILON Synapse with a-Si:H weight storage

In the original EPSILON chip the weight voltage was stored as a voltage on a capacitor. In this new synapse design, shown in Figure 2, the a-Si:H resistance is set such that the voltage drop produced by Iset is equivalent to the original weight voltage, Vw, that was stored dynamically on the capacitor.

A new, simpler, synapse, which can be operated from a single +5v supply, was also be included on the ASiTEST2 chip.

4   The MkII synapse

The circuit is shown in Figure 3. The a-Si:H memory is used to store a current, Iasi. This current is subtracted from a zero current, Isy_z, to give a weight current , +/-Iw, which adds or subtracts charge from the activity capacitor, Cact, thus implementing excitation or inhibition respectively.

For the circuit to function correctly we must limit the voltage on the activity capacitor to the range [1.5v,3.5v], to ensure that the transistors mirroring Isy_z and Iasi remain in saturation. As Figure 3 shows, there are few reference signals and the circuit operates from a single +5v power supply rail, in sharp contrast to many earlier analogue neural circuits, including our own.

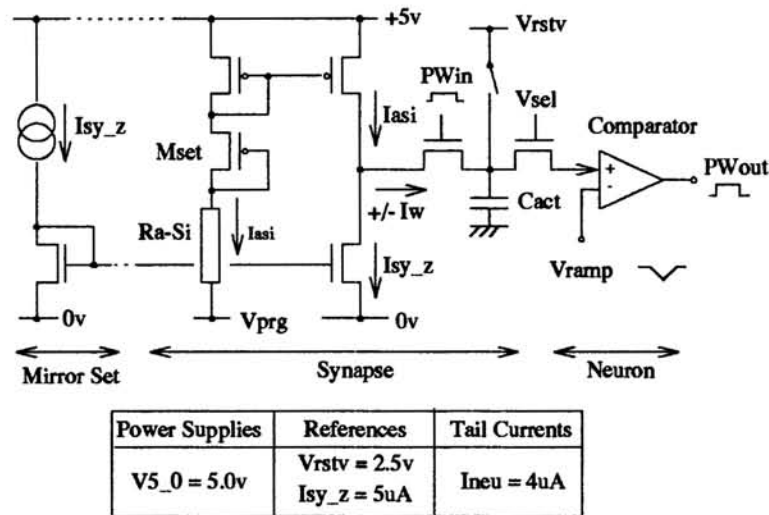| Power Supplies	References	Tail Currents
V5_0 = 5.0v	Vrstv = 2.5v Isy_z = 5uA	Ineu = 4uA

Figure 3: The MkII synapse

On first inspection the main drawback of this design would appear to be a reliance on the accuracy with which the zero current Isy_z is mirrored across an entire chip. The variation in this current means that two cells with the same synapse resistance could produce widely differing values of Iw. However, during programming we do not use the resistance of the a-Si:H device as a target value. We monitor the **voltage on Cact** for a given PWin signal, increasing or decreasing the resistance of the a-Si:H device until the desired **voltage level** is achieved.

Example: To set a weight to be the maximum positive value, we adjust the a-Si resistance until a PWin signal of 5us, the maximum input signal, gives a voltage of 3.5v on the integration capacitor.

We are able to set the synapse weight using the whole integration range of [1.5v,3.5v] by only closing Vsel for the desired synapse during programming. In normal operating mode all four Vsel switches will be closed so that the integration charge is summed over all four local capacitors.

4.1 Example - Stability Test

As an example of the use of integration voltage as means of monitoring the resistance of a particular synapse we have included a stability test. This was carried out on one of the test chips which contained the MkII synapse.

The four synapses on the test chip were programmed to give different levels of activation. The chip was then powered up for 30mins each day during a 7-day period, and the activation levels for each synapse were measured three times.
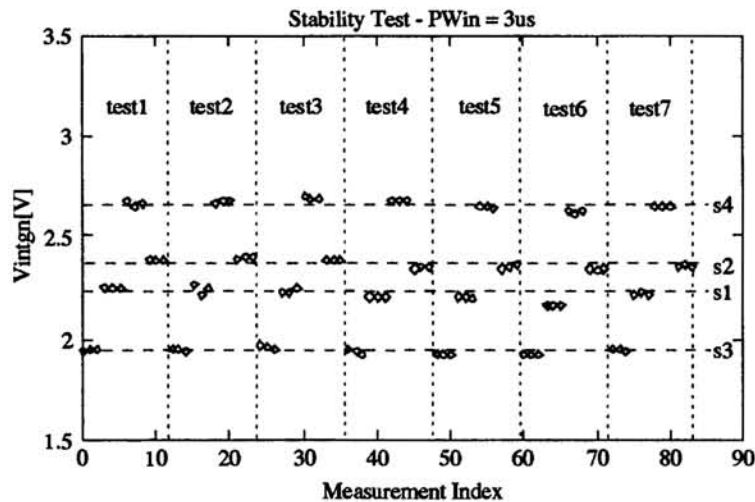

Figure 4: ASiTEST2- Stability Test

As figure 4 shows, the memories remain in the same resistance state (i.e retain their programmed weight value) over the whole 7-day period. Separate experiments on isolated devices indicate much longer hold times - of the order of months at least.

5   ASiTEST3

Recently we have received our latest, overtly neural, a-Si:H based test chip. This contains an 8x8 array of the MkII synapses.

The circuit board for this device has been constructed and partially tested while the ASiTEST3 chips are awaiting the deposition of the a-Si:H layers. We have been able to use an ASiTEST2 chip containing two of the MkII synapse test blocks i.e. 8 synapses and 2 neurons to exercise much of the board's functionality.

The test board contains a simple state machine which has four different states:

- State 0: Load Input Pulsewidths into SRAM from PC.
- State 1: Apply Input Pulsewidth signals to chip1.
- State 2: Use Vramp to generate threshold function for chip1. The resulting Pulsewidth outputs are used as the inputs to chip2, as well as being stored

in SRAM.

- State 3: Use Vramp to generate threshold function for chip2. Read resulting Pulsewidth Outputs into SRAM.
- State 0: Read Output Pulsewidths from SRAM into PC.

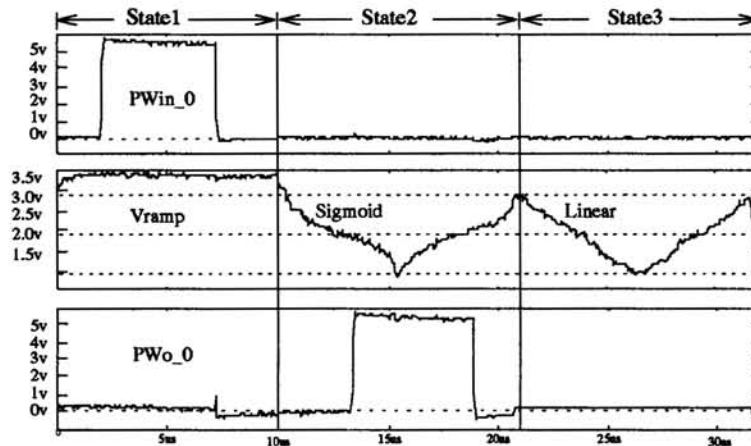The results obtained during a typical test cycle are shown in Figure 5.

Figure 5: ASiTEST3 Board Scope Waveforms

As this figure shows different ramp signals, corresponding to different threshold functions, can be applied to chip1 and chip2 neurons.
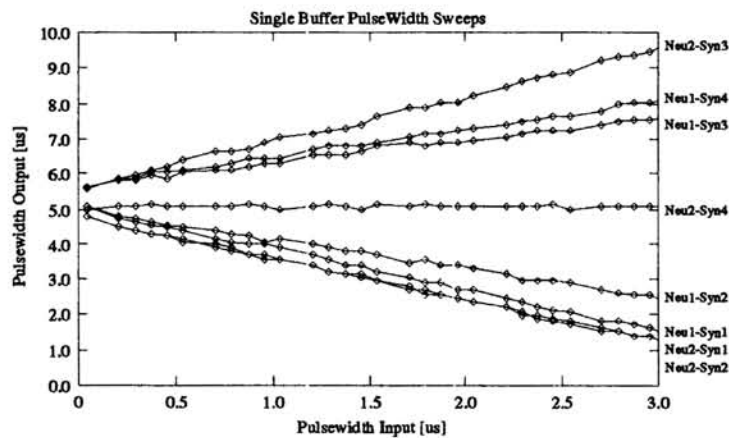

Figure 6: ASiTEST3 Board - MkII Synapse Characteristic

While the signals shown in Figure 5 appear noisy the multiplier characteristic that the chip produces is still admirably linear, as shown in Figure 6. In this experiment all eight synapses on a test chip were programmed into different resistance states and PWin was swept from 0 to 3us.

6 Conclusions

We have demonstrated the use of novel a-Si:H analogue memory devices as a means of storing synaptic weights in a Pulsewidth ANN. We have also demonstrated the operation of an interface board which allows two 8x8 ANN chips, operating as a two layer network, to be controlled by a simple PC interface card.

This technology is most suitable for small networks in, for example, remote control and other embedded-system applications where cost and power considerations favour a single all-inclusive ANN chip with non-volatile, but programmable weights.

Another possible application of this technology is in large networks constructed using Thin Film Technology(TFT). If TFT's were used in place of the CMOS transistors then the area constraint imposed by crystalline silicon would be removed, allowing truly massively parallel networks to be integrated.

In summary - the a-Si:H analogue memory devices described in this paper provide a route to an analogue, non-volatile and fast synaptic weight storage medium. At the present time neither the programming nor storage mechanisms are fully understood making it difficult to compare this new device with more established technologies such as the ubiquitous Floating-Gate EEPROM technique. Current research is focused on firstly, improving the yield on the a-Si:H device which is unacceptably low at present, a demerit that we attribute to imperfections in the a-Si fabrication process and secondly, improving understanding of the device physics and hence the programming and storage mechanisms.

Acknowledgements

References

[1] W. Hubbard et al.(1986) Electronic Neural Networks *AIP Conference Proceedings - Snowbird 1986* :227-234

[2] H.P. Graf (1986) VLSI Implementation of a NN memory with several hundreds of neurons *AIP Conference Proceedings - Snowbird 1986* :182-187.

[3] M.J. Rose et al (1989) Amorphous Silicon Analogue Memory Devices *Journal of Non-Crystalline Solids* 1(115):168-170

[4] A.Hamilton et al. (1992) Integrated Pulse-Stream Neural Networks - Results, Issues and Pointers *IEEE Transactions on N.N.s* 3(3):385-393

[5] M.Holler, S.Tam, H.Castro and R.Benson (1989) An Electrically Trainable ANN with 10240 Floating Gate Synapses. *Int Conf on N.N.s Proc* :191-196

[6] A.F.Murray and A.V.W.Smith.(1987) Asynchronous Arithmetic for VLSI Neural Systems. *Electronics Letters* 23(12):642-643

[7] A.J. Holmes et al. (1993) Use of a-Si:H Memory Devices for Non-volatile Weight Storage in ANNs. *Proc ICAS 15* :817-820