Neuromorphic hardware databases for exploring structure–function relationships in the brain

Catherine Breslin* and Adrian O’Lenskief†

Department of Computing Science and Mathematics, University of Stirling, Stirling FK9 4LA, UK

Neuromorphic hardware is the term used to describe full custom-designed integrated circuits, or silicon ‘chips’, that are the product of neuromorphic engineering—a methodology for the synthesis of biologically inspired elements and systems, such as individual neurons, retinæ, cochleas, oculomotor systems and central pattern generators. We focus on the implementation of neurons and networks of neurons, designed to illuminate structure–function relationships.

Neuromorphic hardware can be constructed with either digital or analogue circuitry or with mixed-signal circuitry—a hybrid of the two. Currently, most examples of this type of hardware are constructed using analogue circuits, in complementary metal-oxide–semiconductor technology. The correspondence between these circuits and neurons, or networks of neurons, can exist at a number of levels. At the lowest level, this correspondence is between membrane ion channels and field-effect transistors. At higher levels, the correspondence is between whole conductances and firing behaviour, and filters and amplifiers, devices found in conventional integrated circuit design. Similarly, neuromorphic engineers can choose to design Hodgkin–Huxley model neurons, or reduced models, such as integrate-and-fire neurons. In addition to the choice of level, there is also choice within the design technique itself; for example, resistive and capacitive properties of the neuronal membrane can be constructed with extrinsic devices, or using the intrinsic properties of the materials from which the transistors themselves are composed. So, silicon neurons can be built, with dendritic, somatic and axonal structures, and endowed with ionic, synaptic and morphological properties. Examples of the structure–function relationships already explored using neuromorphic hardware include correlation detection and direction selectivity.

Establishing a database for this hardware is valuable for two reasons: first, independently of neuroscientific motivations, the field of neuromorphic engineering would benefit greatly from a resource in which circuit designs could be stored in a form appropriate for reuse and re-fabrication. Analogue designers would benefit particularly from such a database, as there are no equivalents to the algorithmic design methods available to designers of digital circuits. Second, and more importantly for the purpose of this theme issue, is the possibility of a database of silicon neuron designs replicating specific neuronal types and morphologies. In the future, it may be possible to use an automated process to translate morphometric data directly into circuit design compatible formats.

The question that needs to be addressed is: what could a neuromorphic hardware database contribute to the wider neuroscientific community that a conventional database could not? One answer is that neuromorphic hardware is expected to provide analogue sensory–motor systems for interfacing the computational power of symbolic, digital systems with the external, analogue environment. It is also expected to contribute to ongoing work in neural–silicon interfaces and prosthetics. Finally, there is a possibility that the use of evolving circuits, using reconfigurable hardware and genetic algorithms, will create an explosion in the number of designs available to the neuroscience community. All this creates the need for a database to be established, and it would be advantageous to set about this while the field is relatively young. This paper outlines a framework for the construction of a neuromorphic hardware database, for use in the biological exploration of structure–function relationships.

Keywords: neuromorphic hardware; silicon neurons; artificial dendrites; evolvable hardware

1. INTRODUCTION

Neuromorphic hardware engineering was added to the repertoire of neuroscience methodologies in the mid-

1980s. Its twofold aims were expounded by one of the founders of the methodology, Mead (1989) as follows: (i) ‘to give computational neuroscience a synthetic element, allowing hypotheses concerning neural organisation to be tested.’, and (ii) ‘To develop an engineering discipline by which collective systems can be designed for specific computations.’ In other words, neuromorphic hardware engineering is a method for both reverse- and

*Author for correspondence.

†Present address: Adobe Systems, Riverside House, 502 Gorgie Road, Edinburgh EH11 3AF, UK.

forward-engineering computational systems, with nervous systems as the exemplar. So far, addressing these aims simultaneously does not appear to be very successful. Attention to reverse-engineering systems often requires the sacrifice of useful (from an engineering perspective) functionality, while attention to forward engineering often takes place at the expense of biological plausibility (Sejnowski *et al.* 1988). We recognize that there is a constant pressure for reverse engineering to feed into forward engineering but, because the theme of this issue is exploring structure–function relationships, we base our discussion on neuromorphic hardware for reverse-engineering purposes. Smith & Hamilton (1998, 1999) contain collections of papers that describe neuromorphic hardware.

Neuromorphic engineers have always emphasized the use of analogue very-large-scale integration circuits, containing between 100 000 and 1 000 000 transistors on a chip (for most examples of neuromorphic hardware, this figure represents capacity, rather than actual device count). Leaving aside more forward engineering concerns, such as the low-power consumption of analogue hardware, the justifications most frequently cited for using analogue hardware, where analogue is defined as computing with continuous-valued, rather than discrete-valued signals, are as follows.

- (i) The analogue nature of signals from the environment. This is important for sensor applications.
- (ii) The analogue and highly nonlinear nature of neural processing, e.g. dendritic processing. This is important for interface and prosthetic applications.
- (iii) The timing of events in analogue hardware is identical to that of biological events. In other words, neuromorphic hardware operates in real-time, as opposed to clocked, or simulated time.
- (iv) The parallel nature of neural processing. It is far easier to achieve parallelism using analogue hardware.
- (v) Arguably, the apparent failure of artificial intelligence to engineer digital nervous systems.

Mead (1989) also advocated the use of analogue hardware because of the concept of a physical equivalence between neurons and field-effect transistors. Transistors operated in the weak-inversion region (below the threshold for conduction of the main population of charge carriers), are subject to the same physical forces as ion channels in the cell membrane, with the consequence that the current between the source and drain terminals of a transistor has the same relationship to the transistor's gate terminal voltage, as the current between extra- and intracellular spaces of a neuron, through a population of ion channels, has to the neuron's membrane potential. The extent to which this is the case, or of any significance, was in dispute and is now ignored. The analogue and highly nonlinear nature of neural processing exists at many different levels of analysis (Bell 1999) and is supported by a vast array of dynamic, physical features. There was a time when neuromorphic engineering was considered to be a methodology for exploring the physical nature of computation, but current practice is to combine pre-designed analogue devices, such as differentiators, amplifiers and filters, so that their transfer function matches

formal descriptions of neurons and networks of neurons, at only one level of analysis. The physical nature of analogue technology itself is massively under-exploited. That said, analogue hardware has been successfully used to construct silicon neurons (Mahowald & Douglas 1991), retinæ (Mahowald 1994), cochleas (Fragniere *et al.* 1997), oculomotor systems (Horiuchi *et al.* 1996) and cortical circuitry (Sarpeshkar *et al.* 2000).

Because neuromorphic engineers are required to construct hardware by modifying, reusing and recombining a collection of circuit devices to which entirely novel designs are rarely added, in practice they tailor the functions of differentiators, amplifiers and filters to match mathematical models of the behaviour of neurons and networks of neurons; there is no systematic nor automated process to assist either the recombination or construction, as there is for digital hardware design. More recently, in recognition of the implications of using analogue hardware, neuromorphic engineers have argued the case for designing hybrid, or mixed-signal hardware, citing the advantages of exploiting the inherent speed and precision of digital computation, and, indeed, the less opaque design process. Of course, the extent to which neural systems can be reverse engineered as a digital phenomenon is still a matter of dispute, and so the majority of neuromorphic hardware remains implemented in analogue hardware.

As an example, suppose a neuromorphic engineer wished to model or reverse engineer the membrane of a neuron. They may choose to model the membrane at a level that exploits the physical equivalence between field-effect transistors and voltage-dependent ion channel-populated membrane. This was described previously. It is articulated clearly by Diorio (1997, p. xi):

'In both integrated circuits and nervous tissue, information is manipulated principally on the basis of charge conservation. In semiconductor electronics, electrons are in thermal equilibrium with their surroundings; their energies are Boltzmann distributed. In nerve tissue, ions are in thermal equilibrium with their surroundings; their energies also are Boltzmann distributed. In semiconductor electronics, we erect barriers to contain the electronic charge, by using the work-function difference between silicon and silicon dioxide, or the energy barrier in a pn junction. The nervous system erects similar energy barriers to contain its electronic charge, by using lipid membranes in an aqueous solution. In both systems, when the height of the energy barrier is modulated, the resulting current flow is an exponential function of the applied voltage. Both systems use this principle to produce devices that exhibit signal gain. Transistors use populations of electrons to change their channel conductance, in much the same way that neurons use populations of ionic channels to change their membrane conductance.'

Modelling at this level has been explored by Breslin & Smith (1999*a,b*) using the structural properties of the materials of the circuitry themselves. This requires a new way of looking at the design of the chip, concentrating on the layout, rather than the circuit schematic level. For example, rather than representing membrane resistance by designing an extrinsic resistor device to be added onto the main circuitry, it exploits the intrinsic resistance of

Figure 1. (a) A population of three delayed-rectifier K^+ channels in a neuronal membrane. The blue helical arrows emerging from the gate represent the time-constant of activation, τ_q . The larger red arrows represent the maximum conductance of the channel, multiplied by the channel density, $G_{\max K^+}$. Finally, the micropipette symbol represents the membrane potential, V_m . (b) The corresponding circuit diagram constructed by matching standard combinations of devices, such as differential pairs, with the transfer functions required to model the membrane equations described in the text. On the right-hand side the circuit diagram the corresponding layout structure is drawn. This is intended only as a guide to facilitate interpretation of (c), which shows a layout plan for the subcell. Various features, required only for fabrication purposes, have been omitted for the sake of clarity. A real layout plan would tend to be more compact than this and with a greater number of wire interconnections. Transistor gates occur where layers of polysilicon intersect with active area. The plates of the capacitor are composed of a layer of thicker polysilicon (polysilicon 2) placed on top of a layer of default polysilicon (polysilicon 1). Transistors M3 and M4, for example, form a differential pair. Resistors can be constructed from lengths of metal or polysilicon. Finally, contacts and vias provide the connections between layers.

the material that composes the membrane structure of the silicon neuron itself. However, it is a level above this at which the majority of neuromorphic engineering work is carried out. Here, the membrane would be represented by collections of differentiators, amplifiers and filters, modelling the kinetic equations of membrane potential behaviour, as described in Hodgkin–Huxley, or reduced Hodgkin–Huxley models. This process is described in

more detail subsequently. Finally, many neuromorphic systems are modelled at a level above membrane kinetics and use either very reduced Hodgkin–Huxley models, such as integrate-and-fire neurons (Glover *et al.* 1998), or omit this level entirely, modelling the behaviour of networks and systems of neurons (Horiuchi *et al.* 1996).

The core structural units for any level of modelling with analogue hardware are field-effect transistors,

capacitors and resistors. The differentiators, amplifiers and filters mentioned previously are composed of small numbers of these units, the sizes and voltage supplies of which can be modified to alter their functionality.

Figure 1 shows the basic elements of a neuromorphic circuit and illustrates how these elements interact between different levels of the neuromorphic design process. The subcell of neuromorphic hardware in this example represents the kinetics of a population of delayed-rectifier voltage-dependent K^+ ion channels according to the Hodgkin–Huxley model. (This subcell is part of a larger circuit described in §2.) Figure 1a depicts a population of these ion channels and the following kinetic variables: τ_q , the time-constant for activation; $G_{\max_{K^+}}$, the maximum conductance of the channel; and V_m , the membrane potential. In figure 1b, which shows the circuit schematic of the subcell, these variables are represented by voltage inputs and outputs. Neuromorphic engineers, in common with analogue engineers, design at the circuit level and tools for simulation operate at this level. In these tools, transistors, capacitors and resistors are represented by built-in models, requiring only the dimensions, voltage supplies and connections between them to be specified. The built-in models range from those composed of ideal current–voltage equations with only important second-order effects included, to semi-empirical approaches that use parameters based on real circuit behaviour. Eventually, these models require parameters from an industrial fabrication process to be used (Weste & Eshraghian 1993).

The circuit is composed of two types of transistors, n-type and p-type (the difference between them being the particles that carry charge in each case), and a capacitor. These devices are combined in such a way as to model the equation for the total conductance associated with the channels:

$$I_{K^+} = (G_{\max_{K^+}} \times q^2)(V - E_{N_{K^+}}), \quad (1.1)$$

where $E_{N_{K^+}}$ is the Nernst potential for K^+ and q is the activation variable with the kinetic form

$$\frac{\delta K^+(t, V)}{\delta t} = \frac{K_{\infty}^+(V) - K^+(t, V)}{\tau_{K^+}(V)}. \quad (1.2)$$

The parts of the circuit that combine to produce each variable are coloured correspondingly. The subcell receives V_m as a voltage input from the upstream circuits; τ_q and $G_{\max_{K^+}}$ are user-controlled voltage inputs. Structurally, the subcell does not resemble a neuronal membrane transversed by voltage-dependent ion channels, nor was it intended to. For further details on this subcell, please refer to Douglas & Mahowald (1995).

For the purpose of the database proposal described in this paper, the most important level of the neuromorphic design process is the level at which the materials that compose the circuit devices are laid out. This is depicted in figure 1c. It is likely that, at the heart of any neuromorphic database, there will be files containing layout descriptions. Standard formats for these files, in which the location and dimensions of each rectangular layer of material is described by a set of coordinates, already exist. They are compatible with all design software tools and form the input to all industrial fabrication processes.

A further level is that at which the hardware is interpreted by neuromorphic engineers as being representative of the structure and function of neurons and networks of neurons. This level has implications for database issues of quality control and meta-analysis. Ultimately, however, we consider the layout level to be the foundation of a neuromorphic hardware database, because of its global compatibility and ubiquity through all forms of silicon-based technologies.

Finally, files containing the location and dimensions of each layer of material are submitted to a design house for fabrication of the hardware itself. For a description of the basics of semiconductor manufacturing, see Weste & Eshraghian (1993, ch. 3).

The remainder of this paper is organized as follows. We first address some aspects of the design, simulation and fabrication of neuromorphic hardware, and discuss the structure–function relationships that it can exhibit. This is necessary for an appreciation of neuromorphic hardware database issues. Then we outline a framework for a neuromorphic hardware database and discuss likely developments in neuromorphic engineering. We conclude by addressing the question: what can a neuromorphic hardware database contribute to the wider neuroscientific community?

2. NEUROMORPHIC DESIGN, SIMULATION AND FABRICATION PROCESS

We have already mentioned how, because of the continuous-valued nature of signals in analogue hardware, there is no algebraic procedure for design, and that the majority of analogue design proceeds with reference to textbooks ('cookbooks') and other sources of pre-designed and characterized circuits. Likewise, there has been no large-scale, concerted effort to construct a database of analogue circuits. This makes it extremely difficult for researchers that lack a background in analogue design to break into the field. Despite the claims of some neuromorphic engineers that the field is exactly about biological inspiration, the vast majority of practitioners are electrical engineers. Unless this situation changes, neuromorphic engineering will inevitably drift towards being an exclusively forward-engineering endeavour. A neuromorphic hardware database would provide an entry point for neuroscientists by interpreting designs in terms of neurobiological structure and function, using familiar terminology. During the design process, circuits are simulated over a range of input–output values, and transient, DC and AC analyses can be performed. Database users can have access to circuit simulation files, with the option of making modifications to a restricted set of parameter values. These simulation tools are the foundation of many early compartmental modelling tools (Gray & Meyer 1993). For the most basic modifications, the only user requirement would be to become accustomed to plausible voltage values. Despite their neurobiological interpretation, most neuromorphic hardware does not operate at neurophysiologically plausible electrical values.

When the design has been completed, the circuit schematic is translated into layout, as illustrated in figure 1c. The only reason for a user to have access to this level would be to modify the dimensions of transistors,

capacitors and resistors, which requires knowledge of device physics. Furthermore, significant changes in device dimensions might necessitate modifications in the layout of surrounding areas to ensure that design rules remain unbroken. When the layout has been completed, an automatic process is carried out to extract a circuit diagram from the layout. This is then compared with the circuit diagram produced earlier in the design process, to identify any mismatch. The layout is also extracted in a format used by the chip foundries in the fabrication of the circuit. If the database user intends to re-fabricate the hardware, they are advised to use the fabrication process¹ and semiconductor vendor² to which the original design was submitted. Although all neuromorphic hardware is fabricated in complementary metal-oxide-semiconductor (CMOS) technology, there are different vendors of, and processes within, the technology.

We therefore anticipate that while experienced analogue designers may choose to make modifications at several levels to which the database provides access, other users will not modify parameters beyond those that are subsequently external to the fabricated hardware.

(a) *Structure–function relationships in neuromorphic hardware*

Though designing hardware with the ability to predict the results of (or at least inform) physiological experiments is an ideal outcome for neuromorphic engineering, its relative youth as a methodology means that it is acceptable to design hardware to demonstrate the feasibility of implementing a particular structure–function relationship for example, or of using a novel design technique.

We wish to present examples of neuromorphic hardware for inclusion in a database. This, in itself, causes problems owing to the existence of a level at which the hardware is interpreted by neuromorphic engineers, usually beginning with the designer themselves, as being eligible for neuromorphic status. One solution is for the database proposer or manager to accept all claims to neuromorphic status without further examination. Another solution is to accept only hardware that satisfies certain constraints. The solution for the purpose of this paper is to use as examples only hardware that is likely to contribute to the study of structure and function in neurobiology. However, neurobiologists should note that this is not a justification favoured by many of the engineers working in the field and that it is extremely likely that a large-scale database will never be constructed for this purpose.

First, we present two examples of neuromorphic neurons. Because they represent the same component, they will occupy the same horizontal axis of the proposed database's tree structure. This is discussed subsequently. However, though they represent the same component, they vary significantly in terms of the subcomponents emphasized and in the design techniques used in their construction. Subcomponents occupy points that are lower down on the same branch. The design technique used is strictly speaking not the concern of the database user, although it may be useful to store this information. It is important, however, to record the fabrication process and vendor used, should the user wish to re-fabricate the hardware or combine it with hardware designed by a

Figure 2. Diagram of the artificial dendritic tree neuromorph. The random input trains A and B are delivered to excitatory synapses on separate branches and to inhibitory branches on a third, common branch. The input trains are uncorrelated for the first 60 ms and correlated, with a degree of random jitter, subsequently. The response to either input alone is negligible because of the sublinear summation of excitatory and inhibitory responses. The response to synchronous input, however, is for the inhibitory responses to sum sublinearly, allowing the excitatory response to propagate. An example of input–output behaviour is shown in the box. The histogram shows output spikes accumulated over 100 trials.

different engineer. These examples also demonstrate the problems of categorizing neuromorphic hardware and illustrate why we have chosen to outline our database in the way we do. We suggest that, unlike many other neurobiological databases, circuit schematics and input–output characteristics should not form the searchable level of the database. We provide abstracted circuit diagrams and input–output characteristics, here, purely as a guide to the form of the hardware. The artificial dendritic tree neuromorph and the silicon neuron neuromorph have been reviewed more extensively by Elias (1993, 1994) and Northmore & Elias (1995), and by Mahowald & Douglas (1991), Douglas & Mahowald (1995) and Rasche *et al.* (1998), respectively, though obviously not in the context of database inclusion. As far as the authors are aware, no comprehensive review of neuromorphic circuits exists in the literature. We present examples of neuromorphic hardware that would occupy other horizontal axes and highlight where vertical links exist.

We begin with the level at which the hardware has been interpreted by the designers.

(i) *Artificial dendritic tree neuromorph*

The artificial dendritic tree neuromorph (Elias 1993, 1994; Northmore & Elias 1995) is a neuron with functionally passive dendrites and a leaky integrate-and-fire model soma (figure 2). It exhibits structure to the extent that the dendrites have various multi-branched configurations and can represent distance, although resistance and capacitance values do not arise from the structure of the dendrites themselves, but are represented with extrinsic devices. It receives excitatory and inhibitory input from synapses that are spatially distributed along the dendrite. Functionally, this neuron hardware exhibits linear or

Figure 3. Diagram of the silicon neuron neuromorph. The circuit receives a current injection input; then for the circuit that represents the K^+ conductance, the membrane potential V_m is low-pass filtered at τ and the output controls the activation transistor of a differential pair, at m . In turn, this is transformed into a voltage that controls the conductance transistor, at G_{K+} . This transistor drains the current from V_m to the K^+ Nernst potential, E_K . With Na^+ , there is additional circuitry for inactivation, u ; again, V_m is low-pass filtered and controls the inactivation differential pair. Activation and inactivation signals combine to give the conductance control voltage. The Ca^{2+} conductance operates in a similar manner. In addition, there is capacitance circuitry that represents intracellular Ca^{2+} concentration.

sublinear summation at the synapses, can be used to discriminate input spike frequency and intervals, and can detect correlation between input trains.

The motivation for the artificial dendritic tree is to model a neuron that has a spatially extensive dendritic tree, so that it has spatio-temporal dynamic signal processing capabilities. This is, in part, to provide an alternative to neural network models that represent the neuron as a point-entity. With reference to its inclusion in a database, we note the following: Elias (Elias 1993, 1994; Northmore & Elias 1995) uses a leaky integrate-and-fire model, implemented in silicon, for a soma. However, other implemented soma models exist, such as the Fitz-Hugh–Nagumo model, implemented by Linares-Barranco *et al.* (1991), or a synapse with memory storage capabilities, such as the device of Diorio *et al.* (1996). These alternative models will occupy the same horizontal axis of the database structure. Likewise, Elias and co-workers experiment with two synapse models that differ in the way in which they achieve weight variation. The first, called a shared conductance array, selects different conductances from an on-chip array, while the second, a self-timed synapse, performs a modulation of the length of time a constant conductance remains activated. As for the soma, these synapse models form subcomponents in the database, together with information on their respective advantages and disadvantages. A further feature of the synapses is their location on the dendritic structure. Elias and co-workers have chosen to place excitatory and inhibitory synapses opposite each other, with uniform distribution. They themselves comment that other distributions may prove more useful. Because the database includes hardware information at the layout level, it

would be possible for an experienced user to alter the distribution of synapses as required. The dendritic structure itself is a series of resistor–capacitor compartments; however, it is configurable to the extent that different branching structures and connection distributions are possible, as are the values in the compartment, which are configurable post-fabrication. Fabrication is originally in a $2\ \mu\text{m}$ CMOS MOSIS (vendor) process.

(ii) *Silicon neuron neuromorph*

We now present an alternative neuron model, for which the emphasis is far more on the ion conductances present at the neuronal membrane. The leaky integrate-and-fire soma used in the artificial dendritic tree neuromorph treats the membrane potential as the result of two processes—a long period of input integration and a very brief period of output pulse production. By contrast, the silicon neuron neuromorph (Mahowald & Douglas 1991; Douglas & Mahowald 1995; Rasche *et al.* 1998) represents the membrane with ion channels described according to the Hodgkin–Huxley model. It represents the following variables: membrane capacitance, resistance and leak conductance; conductance of ion X ($G_{\text{max}X}$), where X is Na^+ or K^+ ; activation variables (q for Na^+ and K^+); inactivation variable (u for Na^+ ; K^+ has an infinite kinetic time variable of inactivation and is therefore not implemented); and time-dependent variables, τ_q and τ_u . Devices are combined to produce the relationships between these variables, as described in equations (1.1) and (1.2). For example, the Na^+ and K^+ conductances have a sigmoidal conductance–voltage relation that is similar to the current–voltage relation generated by a differential pair. The thresholds for activation and inactivation are set by

the gate voltages to one side of the differential pair. The gate voltage for the other side is controlled by the membrane potential. The subcircuit and corresponding layout subcell for the delayed-rectifier K^+ conductance was shown in figure 1. A diagram of the entire circuit is shown in figure 3.

The silicon neuron neuromorph is an example of tailoring the functions of differentiators, amplifiers and filters to match mathematical models of the behaviour of neurons. The motivation was to build a silicon neuron that generated action potentials by modelling the underlying ion conductances. It is possible to configure the somatic conductances to instantiate the characteristics of specific neuronal types; for example, the membrane potential trajectories of neocortical pyramidal neurons are produced by adjusting the parameter settings applied to the various ionic currents. The subcircuits that represent the various ion conductances have a modular design, so that conductances can be included or excluded from designs as necessary and, therefore, exist in a database as independent subcomponents. Work is ongoing (C. Rasche and R. Douglas, unpublished data), to connect functionally active dendrites with the silicon neuron neuromorph. The dendrite exhibits realistic propagation of excitatory postsynaptic potentials, endowing the silicon neuron neuromorph with the properties of antidromic propagation of various firing patterns and direction selectivity. The box in figure 3 shows an example of the input–output characteristics of this circuit, demonstrating direction selectivity by the asymmetrical summation of excitation postsynaptic potentials in the dendrite, from synapses activated sequentially. The fabrication is originally in a $1.2\text{ }\mu\text{m}$ CMOS process. This means that care must be taken if, for example, the user wishes to combine components of this silicon neuron neuromorph with components of the artificial dendritic tree neuromorph, which is fabricated in a $2.4\text{ }\mu\text{m}$ process.

The design of such neurobiologically plausible hardware is occurring less and less frequently. Neurons are more likely to be modelled in extremely reduced forms and incorporated into otherwise forward-engineered systems for sensory processing tasks. An alternative approach is to model a neuron according to its role in a network of neurons and with its implementation in silicon circuitry explicitly in mind. This is demonstrated in a recent paper from the designers of the original silicon neuron neuromorph (Sarpeshkar *et al.* 2000). The motivation was to model a hybrid operation thought to occur in the cortex, in which digital selection of a set of neurons responding to a stimulus is followed by an analogue amplification of this response. This cortex-inspired silicon circuit would, therefore, occupy an axis above that of the silicon neuron neuromorph, or of the artificial dendritic tree neuromorph.

3. A NEUROMORPHIC HARDWARE DATABASE FRAMEWORK

As mentioned previously, independently of neuroscientific motivations, the field of neuromorphic engineering would benefit greatly from a resource in which layout descriptions could be stored for reuse and re-fabrication.

Objections to establishing such a resource in the near future may be made on the basis that there are an insufficient number of neuromorphic hardware designs to warrant full database support. However, it may be worthwhile capturing the existing set of designs in a database if it is anticipated that, at a later stage, the population will become unwieldy. It is also extremely worthwhile to promote discussion of the structure and format of such a resource at a stage when a consensus may be reached. The alternative is to attempt to collate disjoint databases when the field is more mature. In addition, as the field matures, the designs, database entities and relationships between entities will increase in volume and complexity. It is better to explore the entity domain and to evaluate characterization of the entities while there are fewer variables in the system.

In this section, we describe the difficulties that exist in characterizing the entities of the database, before outlining an alternative model to maintain the dataset of neuromorphic designs, and go on to suggest an infrastructure that would support such a model. The primary use of this resource will be to allow a neuromorphic engineer to search and investigate existing designs that exhibit certain structure–function relationships. The difficulty in characterizing the entities of the database is the number of levels at which neuromorphic hardware implements structure–function relationships, for example from neuronal to sensory–motor system, and that the original designer subjectively interprets the hardware as implementing a particular structure. Characterization of the dataset must produce a set of entities that are recognizable and acceptable to all users. We suggest that users should be able to search on the following.

- (i) Component implemented: e.g. ion channel, dendrite, pyramidal neuron, visual interneuron (Harrison & Koch 1999) and outer retinal circuit (Honda *et al.* 1999).
- (ii) Biological plausibility: this remains a subjective interpretation, though attempts have been made to formalize the concept (Webb 1999); however, we feel it would be valuable for the user to have the option of searching on plausibility-related keywords, e.g. reduced, as opposed to Hodgkin–Huxley.
- (iii) Input function: a neuromorphic engineer may wish to search for hardware that can react to a particular input signal. The input signal can, of course, take many forms, with criteria such as spike frequency and voltage level being significant. Neuromorphic hardware that implements sensory–motor systems will often react to real signals, e.g. speech sound (Hussain & Campbell 1999).
- (iv) Output function: a neuromorphic engineer may wish to examine hardware that possesses a particular output characteristic. This would tend to imply forward-engineering motivations, except in cases where the required output will form the input to subsequent stages in a system. The issues that apply to the characterization of input function also apply here.

The subjective, interpreted characteristics of neuromorphic hardware cannot be used to order and structure the database. Initial characterization of the database can

be done on the component modelled by the hardware, organized in a tree-like hierarchy with, for example, cellular and membrane components at the base, and networks of neurons at the top. The primary purpose of this structure is to apply coarse typing of the dataset. A user of the database will be able to search on the vertical axis, the horizontal axis, or in clusters. Searching on the vertical axis allows a particular branch of the database to be searched. A typical use of this would be searching for a function exhibited by a particular implemented component, for example the correlation detection of the artificial dendritic tree neuromorph. The user may find that this function is implemented by a subcomponent of the artificial dendritic tree, or by a more abstract model, of which the artificial dendritic tree is itself a subcomponent, for example in cortical circuitry. Searching on the horizontal axis restricts the level of abstraction. So a user would be restricted to, for example, all neuron neuromorph database entries, and would be searching, for example, for the set of neuron neuromorphs that exhibit certain functions. The ability to search in clusters is an important hybrid of the horizontal and vertical search mechanisms. A typical use of it would be to search implemented components in a set that exhibit a certain function. For example, it might be desirable to search the subcomponents of a neuron on the vertical axis for a particular function and extend the search to the horizontal axis that contains different types of neurons.

The outline, so far, has been concerned with defining groups of entities in which a user may be interested on a coarse scale, with the eventual aim of allowing some search semantics within specific areas of the hierarchy. Each node in the hierarchy contains a list of designs that claim to implement the specific neural element. From each list element, there is a link to the layout description file, and a link to a list of academic publications and reports relating to the design. While the architecture facilitates a coarse search strategy, search mechanisms on academic texts can allow finer-grained searching. The search mechanism would be as powerful as web-based search engines, taking natural language queries. Initially, the aim of the database tool will be only to facilitate the management of the resources.

One of the key issues that will determine the success of the database is deployment. We have identified two major phases in acceptance of the technology: the first is the establishment of an initial population with enough content to attract neuromorphic engineers; the second is the ongoing maintenance and use of the database. The first phase requires a subset of the current research to be categorized and placed in the database; because there is a high degree of subjectivity involved in this task, it will require the cooperation of the original author. In this respect, acceptance of the benefits of such a tool is a prerequisite. When population of the database has reached a certain point, identified by general usage of the tool by the research community, deployment enters the second phase. Unlike the first phase, which placed the onus of initial population of the database on the implementers, the second phase depends on a desire by the research community to add to the database. Although there will be a constant requirement for a database facilitator, the majority of the work required to map the

abstract design onto the database schema would be performed by the submitter, usually the original author.

As the research population is geographically dispersed, irrespective of the back-end database technology used, we believe a web-based front end is a basic requirement. In fact, we view the web browser as being the primary database client. There are many advantages to this approach, not least the fostering of community spirit within the geographically dispersed research groups. Although access to the database is the primary motivation, the portal would also broadcast important information of interest to the field. This would include forthcoming conference details, discussion forums and job postings. The complete range of content available at the portal would be a catalyst for population of the database. Our main interest in the portal at this point is the interaction with the database. We see a need for different classes of users. Guest users have no special access rights, they use the web browser to render views of the database; for example, they might want to browse contents or perform some sort of search. Peer users would have the right to submit new entries for inclusion. Again, we see the web front end as critical in the success of this workflow. Peer users must be identifiable and accountable. There would obviously be administrative users; these are the database facilitators.

With respect to quality control, there are two distinct areas that need to be considered: the database infrastructure and the management of content. The arguments surrounding the quality of different database infrastructure solutions are well understood and the subject of continuous debate. Of interest to us is the quality of the database content. The content, in this case, provides certain characteristics of neuromorphic hardware, together with loose typing by the placement of the database entry in a hierarchy. Before deciding criteria for the quality of content, it is useful to revisit the primary purpose of the database, namely to provide structured access to reference material. In this respect, the database is a course-grained filter; the user would be expected to evaluate the results and verify the material returned. However, as the search mechanics described in this paper require accurate placement of the database entry in the hierarchy, we require the ability to move elements based on current thinking within the research community. Keys for searching include the characteristics captured by the database schema; obviously there is a subjective element to these and the community could play an important part in defining them.

To aid the refinement of the database content, we suggest the application of successive approximation to the dataset. Guest users would have the ability to rate the views returned from the database. This feedback is common in such database applications and is typically used to refine search mechanisms. Peer users would have the ability to manipulate the database entry itself. Of course, with such a dataset, it will not always be possible to reach consensus amongst the community regarding a specific item; therefore some mechanism of generating artificial agreement, possibly using meta-analysis techniques, will be required. In this model, the database represents the body of research contained within the community. The set of characteristics applied to the

dataset by the academic community define the current state of research within that community.

4. DISCUSSION

Function is more accessible to implementation than structure (other than connection structure). Apart from the transformation from three to two dimensions, the barrier to implementing structural features, such as the morphology of neurons, is having to implement resistance and capacitance values using extrinsic devices, such as the switched capacitor. Some ongoing work is addressing the extent to which these characteristics can be obtained from the intrinsic properties of the materials from which circuit elements are composed, allowing the hardware to have a more biologically plausible morphology (Breslin & Smith 1999b).

There are a number of possible developments for neuromorphic engineering. One is the potential use of dynamic, as opposed to static, hardware. The analogue hardware used in neuromorphic engineering, once fabricated, cannot be altered. This is why we have suggested that the layout descriptions linked to database elements should not be modified, because errors in fabricated, modified circuits cannot be fixed. An alternative is to use reconfigurable hardware. Reconfigurable hardware is a method in which hardware, for example field-programmable gate arrays, can be programmed to instantiate different circuit configurations. Essentially, it is reusable hardware, allowing the designer to run experiments with different circuit designs, and giving them a much greater degree of freedom and opportunity for innovation. So far, no reconfigurable hardware dedicated to neuromorphic circuits exists. If an appropriate reconfigurable substrate for instantiating neuromorphic designs is developed, the implications for a neuromorphic hardware database are significant. Users will then be able to search for designs that can be downloaded onto their own copy of the reconfigurable substrate.

A further convolution of reconfigurable hardware techniques is to use them in conjunction with evolutionary algorithms (Thompson 1997). Evolved hardware can incorporate the physical properties of the materials of silicon hardware as integral features of the functional circuits, for example electromagnetic coupling between wires, parasitic capacitances and even flaws on the surface of the silicon itself. This is a genuine solution to the massive under-exploitation of the physical nature of silicon technologies. The implications for a neuromorphic hardware database are profound. It is conceivable that users will be able to define an input–output function and have hardware evolved for them there and then. This takes care of the forward-engineering and functional aspects. The alternative would be for the user to specify the structural components that they wish to form the selectable parts of the evolutionary process.

Finally, we conclude by addressing the question: what can a neuromorphic hardware database contribute to the wider neuroscientific community? Justification for using neuromorphic hardware for modelling in the first place can be found in many places (Breslin 1999; Diorio & Rao 2000; Douglas *et al.* 1995, 1996; Mead 1989; Sarpeshkar 1998), and it is not appropriate to continue this here. We

only intend to begin a discussion on the value of managing this resource with a database. We suggest that the most immediate benefit might occur when neurobiologists wish to place their models into systems that interact directly with the environment and analogue signals from the environment: in other words, robotics applications. Certainly, there is plenty of neuromorphic hardware to populate this region of the database (Patel & DeWeerth 1997; Still & Tilden 1998). A less immediate benefit arises when hardware is required that can interface with real neural processing, for use in prosthetic applications. It is possible to use the electrical activity of a neuron, or other excitable cell, to gate the current of a transistor. Besides being a practical demonstration of a physical equivalence between neurons and transistors, this bidirectional communication is a first step to research into neural prostheses. In addition, the timing of events in neuromorphic hardware is identical to that of neurobiological events. Prostheses will need to be capable of reacting to environmental signals, with timings similar to that of natural nervous systems, and whatever shape an artificial system takes, the degree of parallelism that will be necessary is unlikely to be achieved using purely digital hardware. Hybrid, and possibly evolved, hardware may be required to solve these problems. In the light of the enormity of this task, we hope that promoting discussion of how to manage the components involved will not seem too premature.

END NOTES

¹Processes are defined by the minimum permissible transistor gate width. Currently, the range is *ca.* 2 µm to 0.35 µm. Translating a design from one process size to another is not simply a matter of rescaling the design. Different processes have their own design rules, which can mean that certain conjunctions of layers, or distances between layers permissible in one process, will not be permissible in another.

²Submitting a design to a different vendor may mean that layers in the layout will simply not be recognized. In addition, the more advanced simulation tools will require specific models for each fabrication process and vendor.

The authors wish to thank Christoph Rasche for providing currently unpublished manuscripts and the referees for helpful comments.

REFERENCES

- Bell, A. 1999 Levels and loops: the future of artificial intelligence and neuroscience. *Phil. Trans. R. Soc. Lond. B* **354**, 2013–2020.
- Breslin, C. 1999 Neuromorphic engineering and a successor notion of computation. *New Trends Cognitive Sci.* MIT CogNet (<http://cognet.mit.edu/>).
- Breslin, C. & Smith, L. S. 1999a Neuromorphic design by physical equivalence: simple animal neuron models. In *Third International Conference on Cognitive and Neural Systems*, University of Boston.
- Breslin, C. & Smith, L. S. 1999b Silicon cellular morphology. *Int. J. Neural Syst.* **9**, 491–495.
- Diorio, C. 1997 Neurally inspired silicon learning: from synapse transistors to learning arrays, p. xi. PhD thesis, California Institute of Technology.

- Diorio, C. & Rao, R. 2000 Neural circuits in silicon. *Nature* **405**, 891–892.
- Diorio, C., Hasler, P., Minch, B. & Mead, C. 1996 A single-transistor silicon synapse. *IEEE Trans. Electron Devices* **43**, 1972–1980.
- Douglas, R. & Mahowald, M. 1995 A construction set for silicon neurons. In *An introduction to neural and electronic networks*. New York: Academic Press.
- Douglas, R., Mahowald, M. & Mead, C. 1995 Neuromorphic analogue VLSI. *A. Rev. Neurosci.* **18**, 255–281.
- Douglas, R., Mahowald, M. & Martin, K. 1996 Neuroinformatics as explanatory neuroscience. *Neuroimage* **4**, S25–S28.
- Elias, J. 1993 Artificial dendritic trees. *Neural Comput.* **5**, 648–664.
- Elias, J. 1994 Silicon dendritic trees. In *Silicon implementation of pulse-coded neural networks* (ed. J. Meador), Norwell, MA: Kluwer Academic.
- Fragniere, E., van Schaik, A. & Vittoz, E. 1997 Design of an analog VLSI model of an active cochlea. *Analog Integrated Circuits Signal Process.* **13**, 19–36.
- Glover, M., Hamilton, A. & Smith, L. 1998 An analog VLSI integrate-and-fire neural network for sound segmentation. In *International ICSC Symposium on Neural Computation, Vienna*, pp. 86–89. Millet, Canada: ICSC Academic Press.
- Gray, P. & Meyer, R. 1993 *Analysis and design of analog integrated circuits*. Weinheim; Germany: Wiley.
- Harrison, R. & Koch, C. 1999 An analog VLSI implementation of a visual interneuron. *Int. J. Neural Syst.* **9**, 391–396.
- Honda, A., Kameda, S. & Yagi, T. 1999 Real time image processing with an analog vision chip system. *Int. J. Neural Syst.* **9**, 423–428.
- Horiuchi, T., Morris, T., Koch, C. & De Weerth, S. 1996 Analog VLSI circuits for attention-based, visual tracking. *Adv. Neural Inf. Process. Syst.* **9**, 706–712.
- Hussain, A. & Campbell, D. 1999 Speech-intelligibility improvements using a binaural adaptive-scheme, based conceptually on human auditory processing. *Int. J. Neural Syst.* **9**, 461–466.
- Linares Barranco, B., Sanchez-Sinencio, E., Rodriguez-Vazquez, A. & Huertas, J. 1991 A CMOS implementation of FitzHugh–Nagumo neuron model. *IEEE Solid State Circuits* **26**, 956–965.
- Mahowald, M. 1994 Analog VLSI chip for stereocorrespondence. *IEEE Int. Symp. Circuits Syst.* **6**, 347–350.
- Mahowald, M. & Douglas, R. 1991 A silicon neuron. *Nature* **354**, 515–518.
- Mead, C. 1989 *Analog VLSI and neural systems*. Reading, MA: Addison-Wesley.
- Northmore, D. & Elias, J. 1995 Temporal processing in artificial dendritic tree neuromorphs. In *Proceedings of the 29th Conference on Information Sciences and Systems*, pp. 345–349. Baltimore, MD: Johns Hopkins University.
- Patel, G. & DeWeerth, S. 1997 Analogue VLSI Morris–Lecar neuron. *Electron. Lett.* **33**, 997–998.
- Rasche, C., Douglas, R. & Mahowald, M. 1998 Characterisation of a silicon pyramidal neuron. In *Neuromorphic systems: engineering silicon from neurobiology* (ed. L. Smith & A. Hamilton), pp. 169–177. Singapore: World Scientific.
- Sarpeshkar, R. 1998 Analog versus digital: extrapolating from electronics to neurobiology. *Neural Comput.* **10**, 1601–1638.
- Sarpeshkar, R., Mahowald, M., Douglas, R., Hahnloser, R. & Seung, S. 2000 Digital selection and analogue amplification coexist in a cortex-inspired silicon circuit. *Nature* **405**, 947–951.
- Sejnowski, T., Koch, C. & Churchland, P. 1998 Computational neuroscience. *Science* **241**, 1299–1306.
- Smith, L. & Hamilton, A. (eds) 1998 *Neuromorphic systems: engineering silicon from neurobiology*. Singapore: World Scientific.
- Smith, L. & Hamilton, A. (eds) 1999 *Int. J. Neural Syst.* **9**, issue 5.
- Still, S. & Tilden, M. 1998 Controller for a four legged walking machine. In *Neuromorphic systems: engineering silicon from neurobiology* (ed. L. Smith & A. Hamilton), pp. 177–185. Singapore: World Scientific.
- Thompson, A. 1997 On the automatic design of robust electronics through artificial evolution. In *Proceedings of the 2nd International Conference on Evolvable Systems: From Biology to Hardware*, pp. 13–24. London: Springer.
- Webb, B. 1999 A framework for models of biological behaviour. *Int. J. Neural Syst.* **9**, 375–382.
- Weste, N. & Eshraghian, K. 1993 *Principles of CMOS VLSI design*. Reading, MA: Addison-Wesley.