Analog Computations on Networks of Spiking Neurons

(Extended Abstract)

Wolfgang Maass[1]
Institute for Theoretical Computer Science,
Technische Universitaet Graz,
e-mail: maass@igi.tu-graz.ac.at

NeuroCOLT Technical Report Series

NC-TR-96-014

January 1996[2]

Produced as part of the ESPRIT Working Group
in Neural and Computational Learning,
NeuroCOLT 8556

NeuroCOLT Coordinating Partner

Royal Holloway
University of London

Department of Computer Science
Egham, Surrey TW20 0EX, England

For more information contact John Shawe-Taylor at the above address
or email neurocolt@dcs.rhbnc.ac.uk

[1]Institute for Theoretical Computer Science, Technische Universitaet Graz, Klosterwiesgasse 32/2, A-8010 Graz, Austria. Tel.: +43 316 810063/22, Fax: +43 316 810063/5.
[2]Received December 13, 1995

Abstract

We characterize the class of functions with real-valued input and output which can be computed by networks of spiking neurons with piecewise linear response- and threshold-functions and unlimited timing precision. We show that this class coincides with the class of functions computable by recurrent analog neural nets with piecewise linear activation functions, and with the class of functions computable on a certain type of random access machine (N-RAM) which we introduce in this article. This result is proven via constructive real-time simulations. Hence it provides in particular a convenient method for constructing networks of spiking neurons that compute a given real-valued function f: it now suffices to write a program for computing f on an N-RAM; that program can be "automatically" transformed into an equivalent network of spiking neurons (by our simulation result).

Finally, one learns from the results of this paper that certain very simple piecewise linear response- and threshold-functions for spiking neurons are *universal*, in the sense that neurons with these particular response- and threshold-functions can simulate networks of spiking neurons with *arbitrary* piecewise linear response- and threshold-functions. The results of this paper also show that certain very simple piecewise linear activation functions are in a corresponding sense universal for recurrent analog neural nets.

Keywords: mathematical models for neural networks, spiking neurons, continuous computational complexity, random access machines, analog neural nets.

We have initiated in Maass, 1994a, 1994b, the investigation of the computational complexity of a relatively simple formal model SNN for spiking neuron networks. The goal of this research is to understand the *principles* of information processing in systems whose "processors" exchange and process information in the form of spike-trains, i.e. in the form of time-differences between certain stereotyped events ("spikes").

Although the model SNN is more "realistic" than all models for biological neural nets whose computational complexity has previously been analyzed, it deliberately sacrifices a large number of more intricate biological details (see e.g. Churchland and Sejnowski, 1992, or Aertsen, 1993) for the sake of mathematical tractability.

Very recently one has also started to build VLSI-chips that communicate and manipulate information in the form of time differences between pulses (Watts, 1994, Murray and Tarassenko, 1994). This approach provides another motivation for the development of a new theory of algorithm design and computational complexity for computations that result from a sequence of elementary operations on "spike-trains".

For the sake of completeness we recall here the precise definition of the SNN-model (see Maass 1994a, 1994b, for a detailed discussion). We set $\mathbf{R}^+ := \{x \in \mathbf{R} : x \geq 0\}$.

Definition of a Spiking Neuron Network (SNN): *An SNN $\mathcal{N}$ consists of a* ***finite*** *directed graph $\langle V, E\rangle$ (we refer to the elements of V as "neurons" and to the elements of E as "synapses"), a subset $V_{in} \subseteq V$ of input neurons, a subset $V_{out} \subseteq V$ of output neurons, a threshold-function $\Theta_v : \boldsymbol{R}^+ \rightarrow \boldsymbol{R} \cup \{\infty\}$ for each neuron $v \in V - V_{in}$, a response-function $\varepsilon_{u,v} : \boldsymbol{R}^+ \rightarrow \boldsymbol{R}$ and a weight-function $w_{u,v} : \boldsymbol{R}^+ \rightarrow \boldsymbol{R}$ for each synapse $\langle u, v\rangle \in E$.*

We assume that the firing of the input neurons $v \in V_{in}$ is determined from outside of $\mathcal{N}$, i.e. the sets $F_v \subseteq \boldsymbol{R}^+$ of firing times ("spike trains") for the neurons $v \in V_{in}$ are given as the input of $\mathcal{N}$.

For a neuron $v \in V - V_{in}$ one defines its set F_v of firing times recursively. The first element of F_v is $\inf\{t \in \boldsymbol{R}^+ : P_v(t) \geq \Theta_v(0)\}$, and for any $s \in F_v$ the next larger element of F_v is $\inf\{t \in \boldsymbol{R}^+ : t > s \text{ and } P_v(t) \geq \Theta_v(t-s)\}$, where the potential function $P_v : \boldsymbol{R}^+ \rightarrow \boldsymbol{R}$ is defined by

$$P_v(t) := 0 + \sum_{u\,:\,\langle u,v\rangle \in E} \; \sum_{s \in F_u\,:\,s<t} w_{u,v}(s) \cdot \varepsilon_{u,v}(t-s) \ . \tag{1}$$

The firing times ("spike trains") F_v of the output neurons $v \in V_{out}$ that result in this way are interpreted as the output of $\mathcal{N}$.

We will always assume that there exists some $\tau_{\mathcal{N}} > 0$ such that $\Theta_v(x) = \infty$ for all $x \in (0, \tau_{\mathcal{N}})$ and all $v \in V - V_{in}$ ("refractory period"). This entails that the sets F_v of firing times are well-defined for all $v \in V - V_{in}$. We assume that real-valued inputs and outputs of an SNN are given in the form of time-differences between pairs of spikes.

In models for *biological neural systems* one assumes that if x time-units have passed since its last firing, the current threshold $\Theta_v(x)$ of a neuron v is "infinite" for $x < \tau_{ref}$ (where τ_{ref} = refractory period of neuron v), and then approaches quite rapidly from above some constant value. A neuron v "fires" (i.e. it sends an "action potential" or "spike" along its axon) when its current membrane potential $P_v(t)$ at the axon hillock exceeds its current threshold Θ_v. $P_v(t)$ is the sum of various postsynaptic potentials $w_{u,v}(s) \cdot \varepsilon_{u,v}(t-s)$. Each of these terms describes an *excitatory* (EPSP) or *inhibitory* (IPSP) *postsynaptic potential* at the axon hillock of neuron v at time t, as a result of a spike that had been generated by a "presynaptic" neuron u at time s, and which has been transmitted through a synapse between both neurons. Recordings of an EPSP typically show a function that has a constant value c (c = resting membrane potential; e.g. $c = -70mV$) for some initial time-interval (reflecting the axonal and synaptic transmission time), then rises to a peak-value, and finally drops back to the same constant value c. An IPSP tends to have the negative shape of an EPSP. For the sake of mathematical simplicity we assume in the SNN-model that the constant initial and final value of all response-functions $\varepsilon_{u,v}$ is equal to 0 (in other words: $\varepsilon_{u,v}$ models the *difference* between a postsynaptic potential and the resting membrane potential c). Different presynaptic neurons u generate postsynaptic potentials of different sizes at the axon hillock of a neuron v, depending on the size, location and current state of the synapse (or synapses) between u and v. This effect is modelled by the weight-factors $w_{u,v}(s)$.

The model SNN that we consider in this article is very closely related to the model that was previously considered by Buhmann and Schulten, 1986, and especially to the *spike response model* of Gerstner, 1991, Gerstner, Ritz, van Hemmen, 1992. Similarly as in Buhmann and Schulten, 1986, we consider in this article only the deterministic case (which corresponds to the limit case $\beta \to \infty$ in the stochastic spike response model of Gerstner et al.). However in contrast to these preceding models we do not fix particular (necessarily somewhat arbitrarily chosen) response- and threshold-functions in our model SNN. Instead, we want to have the possibility to use the SNN-model as a framework for *investigating* the computational power of various *different* response- and threshold-functions. In addition, we would like to make sure that various *different* response- and threshold-functions that are observed in specific biological neural systems are in fact special cases of the response- and threshold-functions in the here considered formal model SNN.

We have shown in Maass, 1994b, that one can build from arbitrary neurons, whose response- and threshold-functions satisfy certain basic assumptions, an SNN that can simulate any Turing machine. However SNN's are strictly more powerful than Turing machines for two reasons:

i) An SNN can receive *real* numbers as input, and give *real* numbers as output (in the form of time-differences between pairs of spikes).

ii) We show that one can construct from any neurons which satisfy some rather weak basic assumptions modules for an SNN that can ADD, SUBTRACT, or COMPARE any two sufficiently small phase-differences, as well as a module for MULTIPLY(β) (multiplication of a phase-difference with an arbitrary constant $\beta > 0$). If such operations are applied to a phase-difference of the form $\sum_{i=1}^{\ell} b_i \cdot 2^{-i-c}$, this will in general affect more than $O(1)$ of the bits $\langle b_1, \ldots, b_\ell \rangle$ that are stored in this phase-difference. In contrast to that, any computation step of a Turing machine affects only $O(1)$ bits of any tape content $\langle b_1, \ldots, b_\ell \rangle$.

It turns out that one can in fact *characterize exactly* the computational power of SNN's with arbitrary piecewise linear response- and threshold-functions. For that purpose we consider arbitrary random access machines (RAM's) with $O(1)$ registers that can store in their registers, use as constants, receive as input, and give as their output arbitrary real numbers of bounded absolute value, and which employ arbitrary finite programs that consist of the instructions ADD, SUBTRACT, COMPARE, MULTIPLY(β) for arbitrary real-valued constants β, HALT. These instructions may involve direct and indirect addressing, as well as conditional jumps. We will use the *unit-cost criterion* (i.e. one unit is charged for each execution of an instruction), and refer to the here described RAM's as N-RAM's (because of their intimate connection to neural nets, as shown by the subsequent Theorem). Obviously this model is closely related to that of Blum, Shub and Smale (Blum et al., 1989). It is easy to see that for boolean valued input it can simulate any Turing machine in real-time (representing finite stack-contents $\langle b_1, \ldots, b_\ell \rangle$ by rational numbers $\sum_{i=1}^{\ell} b_i \cdot 2^{-i-c}$ as phase-differences between two oscillators).

Besides providing a tight upper bound for the computational power of a large class of SNN's in terms of a more "user-friendly" type of computational model (N-RAM's), the following result also establishes a relationship between the computational power of SNN's and that of recurrent *analog* neural nets.

In the latter model no "spikes" or other non-trivial timing-phenomena occur, but the output of a gate consists of the "analog" value of some squashing- or *activation function* that is applied to the weighted sum of its inputs. This output value may be interpreted as the current firing-frequency of a neuron. See e.g. Siegelmann and Sontag, 1992, 1994, or Maass, 1993, for recent results about the computational power of such models.

Theorem: *The following three classes of computational models have the same computational power, in the sense that for any model M from one of these classes one can construct models from each of the other two classes that* *can simulate M in real-time (i.e. each computation step of M can be simulated by a constant number of computation steps on the other machines; where each spike on an SNN is counted as one computation step):*

- *SNN's of finite size with piecewise linear response- and threshold-functions and time-invariant weights*
- *recurrent analog neural nets of finite size with piecewise linear activation functions*
- *N-RAM's.*

This equivalence holds both for the case of arbitrary real-valued parameters respectively constants in all three types of models, and if all parameters and constants are required to be rationals.

The **proof** of the preceding result is rather long and complicated. The most difficult part of this proof is the construction of modules for an SNN that can execute the instructions ADD, SUBTRACT, COMPARE, MULTIPLY(β) on real numbers which are represented by phase-differences between two oscillators.

The construction shows in particular that any type of piecewise linear response- and threshold-function that satisfies our *basic assumptions* from Maass, 1994b, is *universal* for *all* piecewise linear response- and threshold-functions, in the sense that *any* SNN with *arbitrary* piecewise linear response- and threshold functions can be simulated in real-time by an SNN with response- and threshold-functions of that particular type. The abovementioned *basic assumptions* mainly require that EPSP's have some (arbitrarily short) segment where they increase linearly, and some (arbitrarily short) segment where they decrease linearly.

The proof of the preceding Theorem also shows that the activation-functions π (saturated linear function) and $\mathcal{H}$ (heaviside) together are in an analogous sense *universal* for all piecewise linear activation functions for recurrent analog neural nets.

An interesting consequence of the proof of the preceding result is that SNN's with piecewise linear *continuous* response-functions are computationally equivalent to recurrent analog neural nets with *arbitrary* piecewise linear activation functions (that may be *discontinuous*). We also would like to point out that N-RAM's with the additional instruction MULTIPLY (for two arbitrary real-valued operands of bounded absolute value) are with regard to real-time simulations equivalent to recurrent analog neural nets with arbitrary piecewise *polynomial* activation functions. It should be noted that Siegelmann and Sontag, 1994, and Koiran, 1993, had already established before some other relationships between analog neural nets and variations of the model by Blum et al., 1989.

Finally, we would like to mention that SNN's with arbitrary piecewise linear response- and threshold-functions and *time-dependent* weights (as specified in Maass, 1994a) are computationally equivalent to N-RAM's with the additional instructions MULTIPLY and DIVIDE for any two real numbers of bounded absolute value.

Acknowledgement

I would like to thank Wulfram Gerstner and Eduardo D. Sontag for helpful discussions.

References

A. Aertsen. ed. (1993) Brain Theory: Spatio-Temporal Aspects of Brain Function. *Elsevier.*

L. Blum, M. Shub, S. Smale. (1989) On a theory of computation and complexity over the real numbers: NP-completeness, recursive functions and universal machines. *Bull. Amer. Math. Soc.* 21(1): 1-46.

J. Buhmann, K. Schulten. (1986) Associative recognition and storage in a model network of physiological neurons. *Biol. Cybern.* 54: 319-335.

P. S. Churchland, T. J. Sejnowski. (1992) The Computational Brain. *MIT-Press.*

W. Gerstner. (1991) Associative memory in a network of "biological" neurons. *Advances in Neural Information Processing Systems, vol. 3, Morgan Kaufmann*: 84-90.

W. Gerstner, R. Ritz, J. L. van Hemmen. (1992) A biologically motivated and analytically soluble model of collective oscillations in the cortex. *Biol. Cybern.* 68: 363-374.

P. Koiran. (1993) A weak version of the Blum, Shub, Smale model. *Proc. of the 34th Annual IEEE Symp. on Found. of Comp. Sci., IEEE Computer Society Press*: 486-495.

W. Maass. (1993) Bounds for the computational power and learning complexity of analog neural nets. *Proc. 25th Annual ACM Symposium on the Theory of Computing*: 335-344.

W. Maass. (1994a) On the computational complexity of networks of spiking neurons (extended abstract). *Advances in Neural Information Processing Systems, vol. 7, Morgan Kaufmann*, to appear.

W. Maass. (1994b) Lower bounds for the computational power of networks of spiking neurons. *Neural Computation*, to appear.

A. Murray, L. Tarassenko. (1994) Analogue Neural VLSI: A Pulse Stream Approach. *Chapman & Hall.*

H. T. Siegelmann, E. D. Sontag. (1992) On the computational power of neural nets. *Proc. 5th ACM-Workshop on Computational Learning Theory*: 440-449.

H. T. Siegelmann, E. D. Sontag. (1994) Analog computation via neural networks. *Theoretical Computer Science* 131: 331-360.

L. Watts. (1994) Event-driven simulation of networks of spiking neurons. *Advances in Neural Information Processing Systems, vol. 6, Morgan Kaufmann*: 927-934.