Robust State-Dependent Computation in Neuromorphic Electronic Systems

Dongchen Liang[1,2] and Giacomo Indiveri[1]
[1]Institute of Neuroinformatics, University of Zurich and ETH Zurich, Switzerland
[2]College of Computer, National University of Defense Technology, Changsha, Hunan, China
Email: [dongchen|giacomo]@ini.uzh.ch

***Abstract*—State-dependent computation is one of the main signatures of cognition. Recently, it has been shown how it can be used as a computational primitive in spiking neural networks for constructing complex cognitive behaviors in neuromorphic agents. However, to achieve the desired computations and behaviors in mixed signal analog-digital neuromorphic electronic systems, these computational primitives should be able to cope with noisy and imprecise components, such as silicon neurons and synapses, with noisy and unreliable external signals, and with interference from the environment. Here we present a spiking neural network model that addresses all these issues while exhibiting both analog signal processing properties and digital symbolic computational abilities. We show how this Neural State Machine (NSM) model can be used for realizing robust state-dependent computation on neuromorphic hardware, and we validate it with experimental results obtained from a recently developed multi-neuron multi-core neuromorphic computing architecture.**

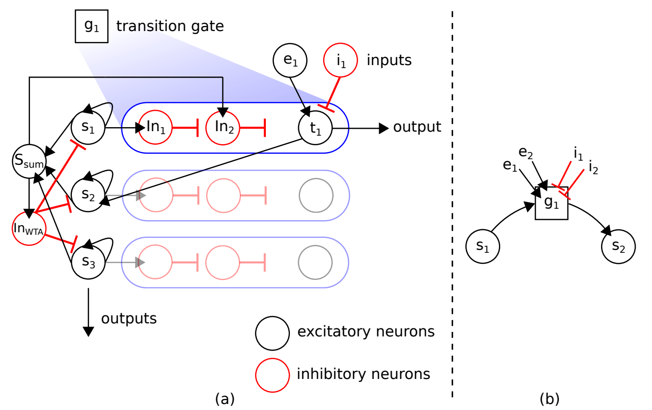

Fig. 1: Neural State Machine. (a) Network structure. (b) Schematic representation.

I. Introduction

State-dependent computation is a hallmark of cognition [1]: external inputs and internal memory collaboratively determine the response of subjects. At the same time, state-dependent computation in finite-state automata is one of the most important computational models in the theory of computation. A variety of complex computations can be developed based on this simple mechanism. In spiking neural networks, Neural State Machines (NSMs) provide a generic computational model for implementing state-dependent computations [2].

Different from von Neumann processor architectures, neuromorphic processors comprise populations of spiking neurons implemented using a combination of slow, low-power, subthreshold analog circuits, and fast programmable asynchronous digital circuits [3]. In this paper, we demonstrate NSMs implementations using a recently developed multi-core neuromorphic processor that integrates 1k analog neurons and 64k digital synapses [4], and that can be configured such that any neuron can send spikes to any other neuron on the same or on other chips. Unlike approaches using clocks or temporal logic, asynchronous neuromorphic systems process data and transmit signals only if and when events (spikes) are produced. Similar to what is observed in biology [5], analog neurons are affected by the variance in their physical substrate and therefore have an amount of mismatch that makes the design of accurate and reliable computing systems challenging. However, biological neural processing systems are an existence proof that reliable and robust computation is indeed achievable using these computational primitives. Here we show how it is possible to achieve equivalent levels of stability and robustness in analog/digital neuromorphic electronic systems by exploiting the dynamics of spiking neuron populations, appropriately connected to each other.

NSMs typically transition from one state to another when both the internal state and external signals are present at the same time. In previous works [2], [6], prescriptions were given for enabling these transitions by putting a threshold on the sum of the activities, and triggering the transition when the sum was above some threshold. This however only works if the activities of the state neuron populations and sensory input populations are in well-defined ranges and if the noise and variability levels are much smaller than the signals. Unsurprisingly, neuromorphic hardware implementations of such models proved to be very sensitive to noise and difficult to control. In this work, we introduce a dis-inhibition mechanism that decouples the parameters of spiking silicon neurons so that their values are no longer restricted to a limited range. With this model, the parameters that determine the neuron and network properties can be configured to have a tolerance distance to the boundary condition required by the state-dependent behaviors. This provides a mechanism that is robust to noise and fabrication mismatch of analog neurons affecting the behavior of NSMs, which emerges out of the network dynamics and does not require the use of large populations of neurons, or careful calibration procedures as in previous cases [2]. In the next section, we describe the spiking neural architecture that gives rise to NSMs and show how they can be implemented with neuromorphic spiking neural network chips. In Section III, we analyze their dynamics and present experimental results measured from the chips that are consistent with the theoretical predictions. In Section IV, we present experimental results from the neuromorphic hardware, and in Section V we present the concluding remarks.

II. Neural State Machines

Similar to finite-state automata, NSMs are defined by four sets of Boolean variables, which include a set of states $S = \{s_1, s_2, s_3 \ldots\}$, a set of excitatory input signals $E = \{e_1, e_2, e_3 \ldots\}$, a set of inhibitory input signals $I = \{i_1, i_2, i_3 \ldots\}$, and a set of transition signals $T = \{t_1, t_2, t_3 \ldots\}$. Each transition of T links one source state of S to at least one target state of S and is also linked to an additional set of input signals. The co-occurrence of input signals and source state signals are used to switch transitions on and off. In particular, we can calculate the external input signal that is sufficient to either start or stop a transition (e.g., t_i) by considering, for all potential inputs $E' \subseteq E$ and $I' \subseteq I$ to the transition population, the two dis-junctions of inputs $e = \vee_j e_j$ and $i = \vee_j i_j$, where $e_j \in E'$ and $i_j \in I'$. The final valid input for the transition is then $v = e \wedge \neg i$. The transition is triggered only if both the source state (e.g., s_i) and v are true, otherwise, it is inhibited. This condition can be expressed as $t_i = v \wedge s_i$. This derivation shows how the inhibition has a stronger effect than excitation. The binary, symbolic, behavior that emerges is based on the assumption that the transition periods are much shorter than the interval of inputs. This assumption can be met by appropriately setting the synaptic and neural time constants in the network. Specifically, to realize the state-dependent computation with spiking neurons, we encode state, input, and transition variables with populations of neurons recurrently connected among each other as illustrated in Fig. 1(a). With the current hardware setup, we use 8 silicon neurons for each population and a shared probability $p \in (0, 1]$ for all the connections. The state populations of S compete among each other in a soft Winner-Take-All (WTA) network. Once a state population wins the competition, its recurrent connectivity self-sustains its activity. The inputs coming from the transition populations t_i can bias the competition by activating other state populations, which would then suppress the currently winning one and compete with each other to become the new winner. In Fig. 1(a), we denote as "transition gate" the structure that determines which transition population t_i can transfer the input to the next states. Only the transition gate g_i connected to the winning state population s_i is open, and all the other gates are actively suppressed through an inhibitory population that is driven by the global activity of the state neurons (through S_{sum} in Fig. 1(a)). This active suppression to close the gates and dis-inhibition to open the desired one is the mechanism that allows us to realize extremely robust state-dependent computation in NSMs. For example, in Fig. 1(a), when s_1 is the winner, t_1 will not be inhibited any more due to the dis-inhibition between In_1 and In_2, which means the transition gate g_1 is open. Once g_1 is open, if i_1 is not given, e_1 can be passed to s_2 by activating t_1. Therefore, transition populations act as a lookup table in which the co-activation of the corresponding state and input neurons enable the activation of the next state population in the network. Fig. 1(b) shows a simplified diagram that represents the full NSM network depicted above.

III. Robustness conditions

We use a mean firing rate model [2], [7] to analyze the dynamics of the NSMs. To simplify the analysis, we make use of a single population of neurons that represents the combination of the populations in Fig. 1(a) (i.e., S_{sum}, In_{WTA} and In_2, or s_1 and In_1), which can both excite or inhibit the state neurons and the transition gate neurons. Under these conditions, the dynamics of a WTA network is described as:

$$\tau \dot{\nu}_{s_n} + \nu_{s_n} = f(\omega_{ts}\nu_t + \omega_{ss}\nu_{s_n} - \omega_{is}\nu_i - T_s) \quad (1)$$
$$\tau \dot{\nu}_{s_c} + \nu_{s_c} = f(\omega_{ss}\nu_{s_c} - \omega_{is}\nu_i - T_s) \quad (2)$$
$$\tau \dot{\nu}_i + \nu_i = f(\omega_{si}\nu_{s_n} + \omega_{si}\nu_{s_c} - T_i) \quad (3)$$

Where $f(\cdot)$ is the half-wave rectification function $max(\cdot, 0)$, ν is the average firing rate of the population neurons, s, t and i represent state, transition and inhibitory populations, and c and n indicate the current and the next state respectively. The term τ is equal to the synaptic time constant. The term ω represents the product of a single synapse weight factor with the number of potential neurons in a population and with the connection probability (e.g., 0.9). The input activity ν_t is determined by the dynamics of transition gates, which are described as:

$$\tau \dot{\nu}_t + \nu_t = f(\omega_{bt} b(t) - \omega_{ii} f(\omega_{it}\nu_i - \omega_{st}\nu_{s_c} - T_t) - T_t) \quad (4)$$

Where $b(t)$ denotes the frequency of external input stimuli, and ω_{ii} is the factor for dis-inhibition (e.g., between In_1 and In_2 in Fig. 1(a)). It shows that ν_t is controlled by the average firing rate of the current state neurons and $b(t)$, which represents the behavior of transition populations (e.g., t_1 in Fig. 1).

In the following analysis, we derive the steady-state solution of the model, which can also approximate the dynamics if we assume that the model time constants are faster than the changes in the inputs [7]. This approximation, however, will not affect the robustness features of the NSM.

A. Steady state of working memory

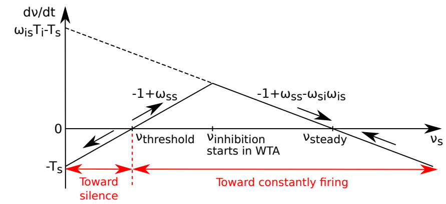

Fig. 2: Fluctuation range of population activity required to maintain the steady state, in WTA networks. Solid lines illustrate the relationship between neurons mean firing rate and its differential when In_{WTA} is turned off and on respectively. To ensure that each state population is a bistable dynamic system, the projection of their intersection onto X-axis must be located between $\nu_{threshold}$ and ν_{steady}. This condition defines (5), (6), (7) and (8).

Given the properties of the WTA network, at steady state, there will always be one winning state population (e.g., s_c), with constant activity, and all other populations will be silent. To self-sustain the activity for the winning state neurons as working memory, the parameters of the soft WTA network should fulfill the following conditions (derived from Fig. 2):

$$\omega_{is}T_i - T_s > 0 \quad (5)$$
$$-1 + \omega_{ss} - \omega_{si}\omega_{is} < 0 \quad (6)$$
$$-1 + \omega_{ss} > 0 \quad (7)$$
$$\omega_{ss}T_i - T_i - \omega_{si}T_s > 0 \quad (8)$$

Each state population can be modeled as a bistable dynamic system with three equilibrium points. Two minima locate where its average firing rate ν_s is equal to 0 and ν_{steady} respectively, and the maximum locates at $\nu_{threshold}$ (see Fig. 2). If the fluctuations of ν_s do not make it cross $\nu_{threshold}$ then it will be sustained to the same average value (see data in Fig. 3 for $t > 2s$). By combining multiple bistable state populations, we can create robust multistable NSMs systems. For all these bistable and multistable systems, ν_t (see (1)) provides the energy to penetrate the barrier (i.e., trigger state transitions). Nevertheless, the settings of all the parameters above are not restricted by the dynamics of state transitions. Based on the conditions above, derived from (1), (2), (3) and (4) when $\nu_{s_n} = 0$, $b(t) = 0$, and knowing that at the steady state, all the differentials are equal to 0, the average firing rate of the winning neurons is given by:

$$\nu_{steady} = \nu_{s_c} = \frac{\omega_{is} T_i - T_s}{1 - \omega_{ss} + \omega_{si}\omega_{is}} \tag{9}$$

Here ν_{steady} and its fluctuation are assumed to be smaller than the saturation frequency of the neurons, set by their refractory period bias. The convergence threshold (see Fig. 2) for the two steady frequencies (the other one is 0) is defined as

$$\nu_{threshold} = \frac{T_s}{\omega_{ss} - 1} \tag{10}$$

In practice, to ensure that the working memory will be able to maintain its state in a stable regime, it is necessary to maximize the distance between ν_{steady} and $\nu_{threshold}$ as well as the distance between $\nu_{threshold}$ and 0. These distances should always be higher than the fluctuation range of the average firing rate of neurons, which can vary due to neuron dynamics, intrinsic and external noise, temperature, etc.

B. Dynamics of state transitions

Other cases that cause the decrease in robustness are undesired or unsuccessful state transitions. The NSM architecture we propose provides a group of conditions that can be used to avoid both of these two cases. In this section, we keep all the parameters discussed in the previous section constant and determine these extra conditions. To simplify the analysis, assume that all the external inputs provided to the transition population t_i are spike trains with constant mean frequency. When an excitatory input is given to the transition population and there is no inhibitory input, $b(t)$ will be equal to a constant value b in the following discussions.

Consider a transition population t_i. When its corresponding state s_i is not firing (i.e., $\nu_{s_c} = 0$), according to (4), if (11) is satisfied, the network has $\nu_t = 0$, which means that the transition gate is closed by the suppression from the current firing state. This mechanism prevents undesired transitions.

$$\omega_{bt} b - \omega_{ii} f(\omega_{it}\nu_i - T_t) - T_t \leq 0 \tag{11}$$

Conversely, when the state population s_i linked to the transition population t_i is active (i.e., $\nu_{s_c} = \nu_{steady}$), according to (4), if (12) is satisfied, the network has $\nu_t = f(\omega_{bt} b - T_t)$, which means that the corresponding transition gate is completely open and ready for incoming external signals. Then according to (1), (13) ensures the start of a new transition, which means that the next state neurons receive enough inputs to increase their firing rate.

$$\omega_{it}\nu_i - \omega_{st}\nu_{s_c} - T_t \leq 0 \tag{12}$$

$$\tau\dot{\nu}_{s_n} = \omega_{ts} f(\omega_{bt} b - T_t) - \omega_{is}\nu_i - T_s > 0 \tag{13}$$

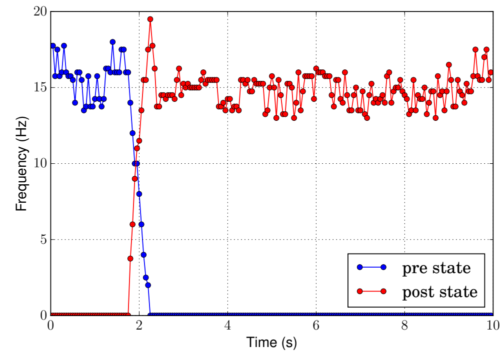

Fig. 3: Dynamics of state transitions measured on silicon neurons from the neuromorphic chip. After the transition at $t = 2s$, the neurons self-sustain their firing rate without any additional external input.

Where $\nu_{s_n} = 0$ due to the previous suppression, and because it is a transient analysis for ν_{s_n}, assume all the other differentials equal to 0 (same for below). According to (13), the network needs a relatively high ω_{bt} to amplify the external input b if its frequency is not high enough to ensure $\omega_{bt} b - T_t > 0$. In addition, according to (11), (12) and (13), with higher ω_{ii}, ω_{st} and ω_{ts}, the network has higher tolerance to the variance of input frequency and the mismatch of silicon neuron circuits.

When the average firing rate of the current state neurons decreases, the frequency of spikes transferred by the transition gate will decrease as well, which means there are fewer events pushing the WTA network to update its winner. To reliably finish this transition, the network should make sure that when the external input is not strong enough to increase the activity of the next state neurons, their average firing rate is still higher than a threshold frequency ν_{equal}, which occurs when two populations have equal chance to win the competition. This threshold value is higher than the ideal value given in (14), which holds for steady state conditions, or conditions in which all the time constants are equal to 0, and there is no noise. It is derived from the ν_{s_n} of (1), (2) and (3) at a steady state when $\nu_{s_c} = \nu_{s_n}$ and $\nu_t = 0$. If there is no fluctuation, ν_{equal} should be smaller than ν_{steady}.

$$\nu_{equal} \geq \frac{\omega_{is} T_i - T_s}{1 - \omega_{ss} + 2\omega_{si}\omega_{is}} \tag{14}$$

To finish the transition, when ν_{s_n} is increased to ν_{equal}, it is required that the transition gate is not totally closed (described in (15)). Also, the network should satisfy the condition given in (16), which means the activity will continue increasing to cross this point. These conditions are derived from (4) and (1) respectively when $\nu_{s_c} = \nu_{s_n} = \nu_{equal}$.

$$\nu_t = f(\omega_{bt} b - \omega_{ii} f(\omega_{it}\nu_i - \omega_{st}\nu_{equal} - T_t) - T_t) > 0 \tag{15}$$

$$\tau\dot{\nu}_{s_n} = \omega_{ts}\nu_t + (\omega_{ss} - 1)\nu_{equal} - \omega_{is}\nu_i - T_s > 0 \tag{16}$$

According to (15) and (16), ω_{st} should be high enough (compared with ω_{it}) to keep the transition gate open. In addition, ω_{ts} needs to be high for transferring the input to the next state neurons. These conditions prevent unsuccessful transitions. The state transition fulfilling all the conditions described above is illustrated in Fig. 3 at $t = 2s$.

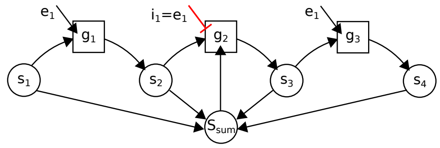

Fig. 4: Network for eliminating ambiguous inputs.

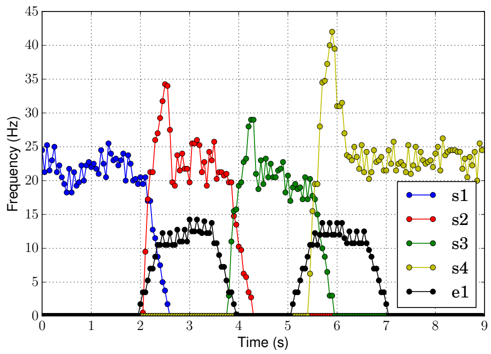

Fig. 5: Consecutive transitions using the same input. Input population e_1 receives 10 Hz constant external stimuli with an interval between $t = 4s$ and $t = 5s$. Because of the interval, the NSM transitions from state s_2 to s_3, then consecutive input from e_1 triggers another transition.

As discussed above, the parameters of NSMs cooperatively define the state-dependent behaviors. However, the tuning of these parameters is not restricted to a narrow range which is correlated to the value of other parameters.

IV. Hardware experiments

We implemented the NSM architecture illustrated in Fig. 1(a) using a multi-core neuromorphic chip [4] and measured mean firing rates from its silicon neurons. We show that in the hardware NSMs, a wide range of probabilities can be used to define the connectivity between populations (e.g., $p = 0.9$ for Fig. 3 and Fig. 5, or $p = 0.5$ for the robustness test). This demonstrates the increased robustness of the architecture proposed, relieving the precision limits from the hardware resources and providing more options when developing applications on neuromorphic hardware.

A. Ambiguous inputs

In addition to mismatch and noise both in the circuits and in the environment, also ambiguous inputs can cause undesired transitions, as also discussed in [2]. Due to the analog circuit mismatch, the system noise, and the analog neural dynamics, it is hard to precisely control the duration and strength of the inputs given to individual neurons. When the same input condition is applied multiple times for state transitions, the network cannot easily distinguish how many transitions should be triggered. In this work, to eliminate this ambiguity, we propose to use a NSM that has a dedicated transition unit to signal the end of an input symbol and a new transition gate to initiate new transitions triggered by the same input symbol (see Fig. 4). In this way, multiple transitions are triggered by the same input symbol only if there is an interval between instances of the applied input signal (see Fig. 5).

TABLE I: Proportion of successful transitions.

	Model in [2]	Neural State Machine
Self-transitions	97.4%	100%
Without ambiguity	95.44%	100%
With ambiguity	88.97%	100%

B. Robustness test

We tested the system with a NSM that has 8 states, 16 inputs and 128 random transitions (97 self-transitions, 24 transitions without ambiguity and 7 transitions with ambiguity). We triggered 10,000 transitions in random order and compared the robustness of our model to previously proposed approaches (see Table I). In the test, each input signal consists of 5 spikes sent with 20 Hz. The inputs are given with 500 ms intervals. During the test, state holding neurons are firing at 25 Hz on average.

V. Conclusions

We proposed a novel NSM architecture that makes use of transition gates to realize robust state-dependent computation in analog/digital neuromorphic circuits. The reliability and robustness features of the model do not require more than 8 neurons/population. This allows us to use one single self-excited soft WTA network to hold and update the memory of internal states, overcoming the requirement of multiple coupled soft WTAs of previous models [2], [6]. In this work, we analyzed the dynamics of NSMs and provided a set of conditions for ensuring their robust behavior. NSMs provide a computational primitive for integrating large-scale asynchronous cognitive computations in neuromorphic electronic systems. In addition to solving practical engineering problems, the proposed architecture might shed light on the role of dis-inhibition in biological networks.

Acknowledgments

This work is supported by the China Scholarship Council and by the Institute of Neuroinformatics, University of Zurich and ETH Zurich.

References

[1] P. Dayan, “Simple substrates for complex cognition,” *Frontiers in Neuroscience*, vol. 2, no. 2, p. 255, 2008.

[2] E. Neftci, J. Binas, U. Rutishauser, E. Chicca, G. Indiveri, and R. Douglas, “Synthesizing cognition in neuromorphic electronic systems,” *Proceedings of the National Academy of Sciences*, vol. 110, no. 37, pp. E3468–E3476, 2013.

[3] E. Chicca, F. Stefanini, C. Bartolozzi, and G. Indiveri, “Neuromorphic electronic circuits for building autonomous cognitive systems,” *Proceedings of the IEEE*, vol. 102, no. 9, pp. 1367–1388, 9 2014.

[4] G. Indiveri, F. Corradi, and N. Qiao, “Neuromorphic architectures for spiking deep neural networks,” in *Electron Devices Meeting (IEDM), 2015 IEEE International*. IEEE, Dec. 2015, pp. 4.2.1–4.2.14.

[5] A.A. Faisal, L.P. Selen, and D.M. Wolpert, “Noise in the nervous system,” in *Nature reviews. Neuroscience*. vol. 9, no. 4, pp. 292, 4 2008.

[6] U. Rutishauser and R. Douglas, “State-dependent computation using coupled recurrent networks,” *Neural Computation*, vol. 21, pp. 478–509, 2009.

[7] P. Dayan and L. Abbott, *Theoretical Neuroscience: Computational and Mathematical Modeling of Neural Systems*. Cambridge, MA, USA: MIT Press, 2001.