Inference in Spiking Bayesian Neurons using Stochastic Computation

Chetan Singh Thakur, Jamal Molin, Ralph Etienne-Cummings
Department of Electrical and Computer Engineering
Johns Hopkins University
Baltimore, MD, USA
Email: cthakur2@jhu.edu

André van Schaik
The MARCS Institute
Western Sydney University
Sydney, NSW, Australia

***Abstract*—We present a stochastic Bayesian neuron (SBN) that codes for a binary hidden variable and the temporal dynamics of which can be explained as a Bayesian inference. We show that our SBN combines the maximum likelihood of its synaptic inputs and the prior probability of the hidden variable to infer the presence of the hidden variable. Probabilistic models are computationally complex, which makes them difficult to implement using standard state-of-the-art digital implementation. Here, we employ stochastic logic elements to implement the SBN using minimum hardware resources. The SBN could be used as a basic element to develop a Bayesian processor that works on probability instead of deterministic logic.**

I. Introduction

Animals are constantly faced with the challenge of estimating external world states using noisy sensors. Various psychophysical experiments suggest that biological neurons estimate Bayesian posterior probabilities of hidden states using observations from noisy sensors [1]–[4]. In recent years, Bayesian models are being increasingly used to explain sensory perception, motor control, and reasoning, based on an estimation of underlying hidden variables from sensory observations [5]–[7]. However, none of the existing neuron models could be used as a basic element to perform Bayesian inference for estimating the hidden variable. An exception is the Bayesian neuron model by Deneve et al. [8] that has been used to explain the underlying theory of neural system processing, but requires computationally expensive arithmetic to implement on hardware for real-time probabilistic processing.

Modern computing hardware is constrained by stringent application requirements such as extremely small size, low power consumption, and high reliability. Further, physical phenomena, such as manufacturing process variations and soft errors, give rise to error-prone behaviour that can be best described in probabilistic terms. Consequently, unconventional computing methods, such as stochastic computation (SC), that directly address these issues are of increasing interest. A basic feature of SC is that numbers are represented by bit-streams that can be processed using simple circuits. The numbers themselves are interpreted as probabilities under both normal and faulty conditions. For example, multiplication can be performed using a stochastic circuit consisting of a single AND gate. Consider two binary bit-streams that are logically ANDed. If the probabilities of seeing a 1 on the input bit-streams are p1 and p2, then the probability of 1 at the output of the AND gate will be p1 × p2, assuming that the two bit-streams are suitably uncorrelated or independent.

Here, we propose a stochastic Bayesian neuron (SBN) that codes for a binary hidden variable and could be useful as a basic building block in probabilistic processors. We show that the SBN can compute complex probabilistic arithmetic and can be implemented using simple digital circuits. We show the hardware implementation of a simple SBN model that calculates the odds ratio of the maximum a posterior of the hidden variable without requiring any normalisation term. Our novel SBN model is inspired from the stochastic theory proposed by Gaines [9] and has the advantage of requiring an extremely small silicon area for implementation. Our approach is robust to soft errors and this will facilitate the development of Bayesian computing machines in nanometer fabrication technologies.

II. Overview of the SBN Model

Our problem formulation of the SBN model is similar to that proposed by Deneve [8]. We consider that each neuron codes for a binary hidden variable *h*. This variable could correspond to a property of the real world such as the presence or absence of an object or the direction of motion in the neuron's receptive field. The hidden variable could also be much more abstract and represent statistical regularities of the sensory input and motor output. The variable is "hidden" from the neuron that tries to infer its state from its synaptic spikes. As an illustrative example, we will consider that *h* represents the presence or absence of a vertical bar at a certain position on the retina. The synaptic inputs become activated by external hidden variables with particular probabilities (Fig. 1).

The synaptic inputs of the SBN can be represented as s^i_t, when the synapse *i* is activated between time *t* and $t + dt$. Here, the synaptic inputs can be described by a Poisson process. The probability of activation of the i^{th} synapse, given the hidden variable *h* is present, is defined as $P(s^i_t \mid h) = g^i_{on}/g^i_{off}$. $P(s^i_t \mid \overline{h}) = g^i_{off}/g^i_{on}$ is the probability of the i^{th} synapse being activated, given the hidden variable *h* is not present (i.e., $\overline{h}$). The synapse *i* would be excitatory if $g^i_{on} > g^i_{off}$, else it would be inhibitory. g^i_{on} and g^i_{off} are the parameters of the synaptic circuit.

We can formulate the inference in the SBN as a Bayesian inference problem in terms of the odds ratio of the hidden variable *h*, given all the synaptic inputs received in the past, using the equation:

978-1-5090-4780-2/17/$31.00 ©2017 IEEE

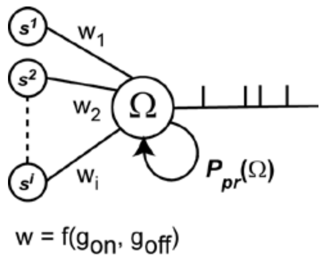

Fig. 1. Model of the stochastic Bayesian neuron (SBN).

$$\Omega_t = P(h \mid S_t)/P(\bar{h} \mid S_t)$$

Using Bayes theorem and considering that all the synaptic inputs are independent to each other,

$$\Omega_t = [P(S_t \mid h) \times P_{pr}(h)] \, / \, [P(S_t \mid \bar{h}) \times P_{pr}(\bar{h})]$$
$$= [P_{pr}(h) \, / \, P_{pr}(\bar{h})] \; \times \prod_{i=1}^{k} [P(s_t^i \mid h) / \, P(s_t^i \mid \bar{h})] \qquad (1)$$

where, Ω_t is the membrane potential of the SBN, S_t is the vector of the binary variables corresponding to k synaptic inputs received by the SBN at time t, $P_{pr}(h)/P_{pr}(\bar{h})$ is the odds ratio of the prior probability of the hidden variable h, and g^i_{on}/g^i_{off} is the synaptic weight of the i^{th} synapse.

III. Hardware Implementation Using SC

A. Synapse Circuit

In Fig. 1, we have described the SBN model, which is responsive to the presence of a vertical bar in its receptive field. Synapses of the SBN calculate the odds ratio of likelihood in the formulation of the Bayesian inference problem as in equation (1). Fig. 2 shows a synapse circuit whose output goes to the SBN as a train of spikes. This synapse circuit is inspired by the ADDIE (ADaptive DIgital Element) circuit proposed by Gaines [9].

The probability $P(S)$ of the output variable S, is $(count/2^d)$, where d is the bit-width of the counter. The dual-line complementary output ($P(S)$ and $P(\overline{S})$) of the synapse circuit will represent the ratio of the two input variables. The detailed derivation of a similar circuit can be found in previous works [9], [10]. The odds ratio for the output of the synapse circuits can be written as:

$$P(S)/P(\overline{S}) = P(S)/(1 - P(S)) = g_{on}/g_{off}, \; if \; h$$
$$= g_{off}/g_{on}, \; if \; \bar{h}$$

In a synapse circuit, the value of N could be as low as 1 (i.e., single flip-flop). The counter circuit is important because it converts the ratio of two independent variables into an odds ratio of a single variable.

B. Neuron Circuit

The membrane potential of the SBN represents the odds ratio of the external hidden variable as defined in equation (1). Using the principles of SC, we can implement the SBN by multiplying the odds ratio of the synaptic inputs with the prior odd of the hidden variable using a simple AND gate. In this example, we have considered three synapses (Fig. 3). $P(s_1)$,

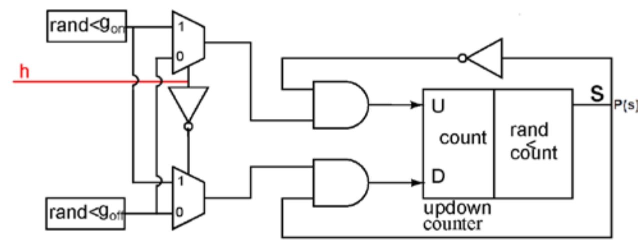

Fig. 2. Synapse circuit in the SBN model. The circuit generates spike trains based on its internal parameters g_{on} and g_{off} as the evidence of the external hidden variable.

$P(\bar{s}_1)$, $P(s_2)$, $P(\bar{s}_2)$, $P(s_3)$, and $P(\bar{s}_3)$ are the outputs of three synapse circuits (Fig. 3), and $P(h)$ and $P(\bar{h})$ are the prior probabilities of the hidden variable. The spiking probability $P(O)$ of the neuron has an underlying inhomogeneous Poisson process and generates a spike based on the instantaneous membrane potential, Ω. We can write the equation of the neuron circuit as:

$$U = P(s_1) \times P(s_2) \times P(s_3) \times P_{pr}(h) \times P(\overline{O})$$

$$D = P(\bar{s}_1) \times P(\bar{s}_2) \times P(\bar{s}_3) \times P_{pr}(\bar{h}) \times P(O)$$

Incoming spikes at terminals U and D represent excitatory and inhibitory spikes, respectively (Fig. 3). The membrane voltage, Ω, is represented as a counter, which acts as an integrator for the incoming excitatory and inhibitory spikes. The counter has a width of d bits. Thus, the maximum state value is $N = 2^d - 1$ and the total states are $N + 1$.

The output spike of the neuron can be written as:

$$P(O) = \int U - D$$

Using the above equations,

$$P(O) = \int [P(S) \times P_{pr}(h) \times P(\overline{O})] - [P(\overline{S}) \times P_{pr}(\bar{h}) \times P(O)]$$

where,

$$P(S) = P(s_1) \times P(s_2) \times P(s_3), \text{ and } P(\overline{S}) = P(\bar{s}_1) \times P(\bar{s}_2) \times P(\bar{s}_3)$$

Rearranging the above equation,

$$P(O)' = dP(O)/dt = [P(S) \times P_{pr}(h) \times P(\overline{O})] - [P(\overline{S}) \times P_{pr}(\bar{h}) \times P(O)]$$

$$P(O)' + [P(\overline{S}) \times P_{pr}(\bar{h}) \times P(O)] = P(S) \times P_{pr}(h) \times P(\overline{O})$$

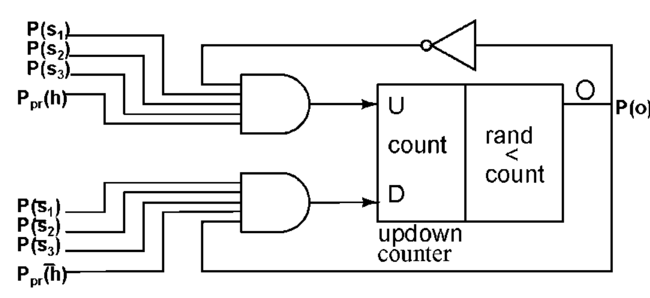

Fig. 3. Neuron circuit in the SBN model. The circuit performs the inference of the external hidden variable using the odd of the likelihood ratio computed by the synapse circuits, and the prior probability of hidden variable.

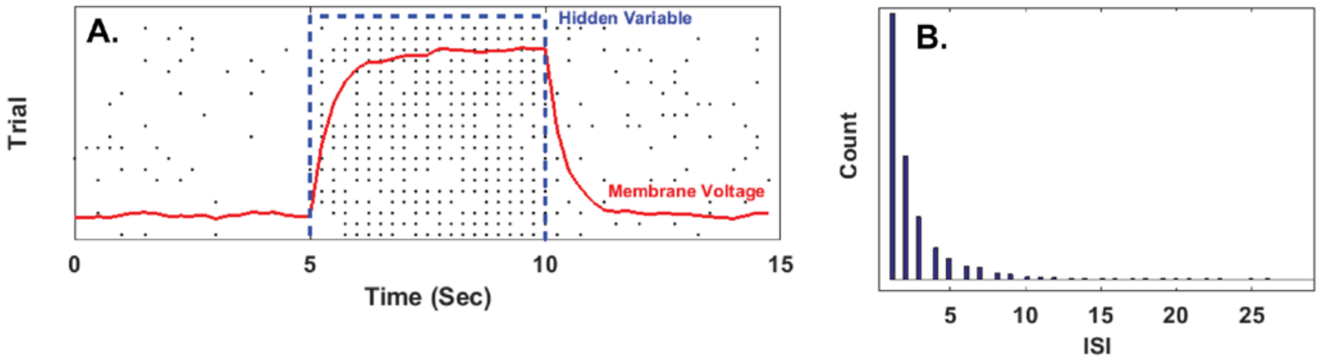

Fig. 4. Characteristics of the SBN. (A) Mean membrane voltage of the SBN across 20 different trials in response to the external hidden variable, (B) Inter-spike interval (ISI) of the output spikes of the SBN for a single trial.

Dividing the above equation by $P(\bar{O})$, and assigning $\Omega = P(O)/P(\bar{O})$, we get:

$$\Omega' + \Omega \times (P(\bar{S}) \times P_{pr}(\bar{h})) = P(S) \times P_{pr}(h) \quad (2)$$

Equation (2) is a first-order differential equation. At equilibrium, the change in the output probability is zero, i.e. $\Omega' = 0$, which will give a stationary solution as in equation (3).

$$\Omega(n) = [P(S) \times P_{pr}(h) \,/\, P(\bar{S}) \times P_{pr}(\bar{h})](1 - e^{-n/N}) \quad (3)$$

where, N is the time constant of the dynamical system and n is the number of discrete time steps.

When $n >> N$, the steady state solution would be:

$$\Omega(n >> N) = P(S) \times P_{pr}(h) \,/\, P(\bar{S}) \times P_{pr}(\bar{h})$$

This shows that the SBN can perform the Bayesian inference of an external hidden variable using a simple circuit. We have explored the dynamics of the SBN in the results section below.

IV. Results

A. Inference by a single neuron

In Fig. 4A, we show the mean membrane voltage (in red) across 20 different trials. It is evident that the membrane potential has higher values when the external hidden variable (in blue) is present. We further plotted the output spikes of the SBN across different trials and found that the spike density is higher in the presence of hidden variable. In this scenario, we have considered three synapses, with g_{on1}/g_{off1} = 0.9/0.5, g_{on2}/g_{off2} = 0.3/0.4, and g_{on3}/g_{off3} = 0.8/0.3. We used 6-bits counter as a membrane capacitance to store the incoming synaptic charges. Fig. 4B shows the inter-spike interval of the output spikes of the SBN for a single scenario. It can be seen that it follows the Poisson distribution, with a rate parameter of 2.53.

In Fig. 5, we demonstrate the dynamics of the membrane potential of the neuron for various time constants, N (equation [3]), for different counter sizes (3-, 5-, and 7-bit). It can be seen that a smaller time constant results in a higher variance in the prediction, though the response time is lower.

B. Cue coupling as a hierarchical inference

We consider a simple example, which assumes a hidden variable (h) causing two cues, visual (V) and auditory (A). We consider three synapses for each cue, with $g_{onV\_i} = g_{onA\_i}$ and $g_{offV\_i} = g_{offA\_i}$. Their values for these examples are g_{on1}/g_{off1} = 0.5/0.2, g_{on2}/g_{off2} = 0.8/0.1, and g_{on3}/g_{off3} = 0.8/0.3. We have used a 7-bit counter as the membrane charge storage. We have considered a strong prior for the visual cue (0.5) compared to

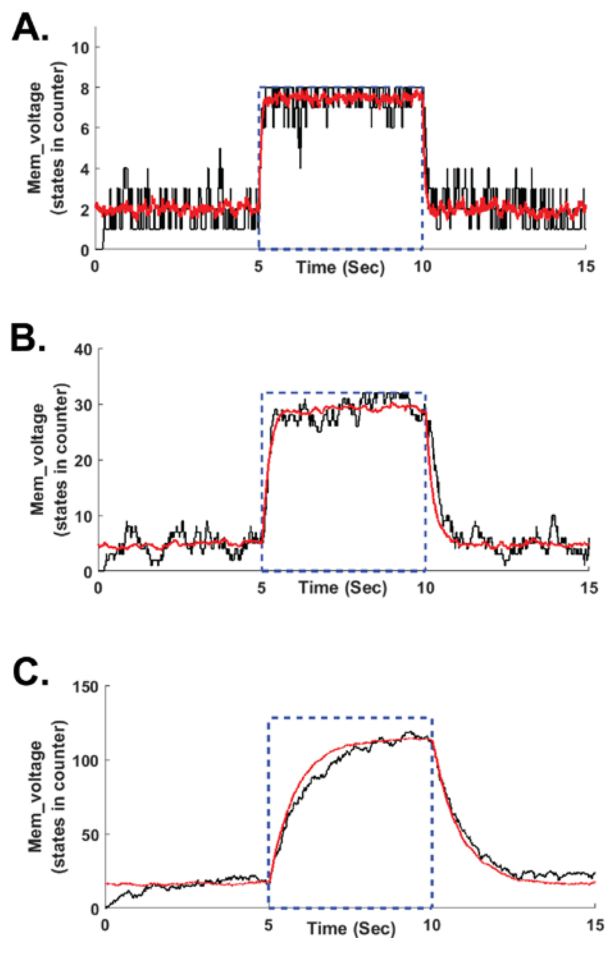

Fig. 5. Evolution of membrane voltage for different time constants, i.e., different counter bit-widths, (A) 3-bit, (B) 5-bit, (C) 7-bit. The blue square indicates the presence of the hidden variable. The red and black curves show the mean membrane voltage across 20 trials and 1 trial, respectively.

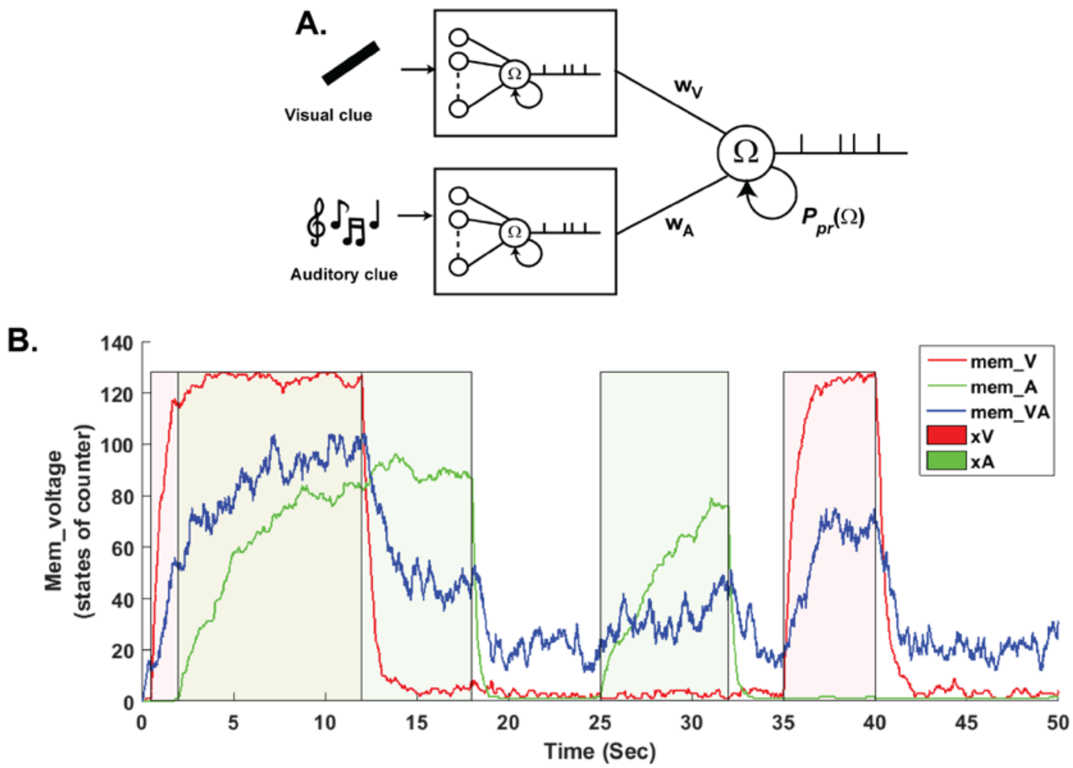

Fig. 6. Simulation of a simple hierarchical Bayesian network. (A) Schematic of the network (B) Evolution of membrane voltage of neurons activated by visual cue (red), auditory cue (green), and combined visual and auditory (blue) cues. Red and green rectangles show the presence of visual (xV) and auditory (xA) hidden variables, respectively, with some overlapping areas.

the auditory cue (0.01). The inference in the final neuron can be written as:

$$\Omega = P(h \mid V,A) / P(\bar{h} \mid V,A)$$
$$= P(V,A|h) \times P_{pr}(h) \,/\, P(V,A|\,\bar{h}) \times P_{pr}(\bar{h})$$

Assuming both cues are independent of each other,

$$\Omega = P(V|h) \times P(A|h) \times P_{pr}(h) \,/\, P(V|\,\bar{h}) \times P(A|\,\bar{h}) \times P_{pr}(\bar{h})$$

As shown in Fig. 6, the membrane voltage of the auditory neuron has a larger time constant compared to that of the visual cue because of a prior, and this makes the auditory neuron less confident about the external hidden variable. The activation of the VA inference neuron is higher when both V and A cues are present as compared to when a single cue is present.

V. Conclusions

In this paper, we have presented a Bayesian spiking neuron model. This is one of the first neuron models to perform Bayesian inference, which can be implemented using a very simple circuit. We have shown the dynamical behaviour of this neuron model, and the manner in which it can be used to build a larger network to perform hierarchical Bayesian inference. Future work will include incorporation of learning of the model parameters. We envision that this neuron model will serve as a basic building block to develop Bayesian processors, which will work on probabilistic arithmetic rather than deterministic logic.

References

[1] D. E. Angelaki, Y. Gu, and G. C. DeAngelis, “Multisensory integration: psychophysics, neurophysiology, and computation,” *Current Opinion in Neurobiology*, vol. 19, no. 4, pp. 452–458, Aug. 2009.

[2] T. Lochmann and S. Deneve, “Neural processing as causal inference,” *Current Opinion in Neurobiology*, vol. 21, no. 5, pp. 774–781, Oct. 2011.

[3] M. G. Paulin and A. van Schaik, “Bayesian Inference with Spiking Neurons,” *Neural Computation*, pp. 1–26, Jun. 2014.

[4] M. G. Paulin and L. F. Hoffman, “Bayesian head state prediction: Computing the dynamic prior with spiking neurons,” in *2011 Seventh International Conference on Natural Computation*, 2011, vol. 1, pp. 445–449.

[5] K. Körding, “Decision theory: what ‘should’ the nervous system do?,” *Science (New York, N.Y.)*, vol. 318, no. 5850, pp. 606–10, 2007.

[6] M. O. Ernst and M. S. Banks, “Humans integrate visual and haptic information in a statistically optimal fashion.,” *Nature*, vol. 415, no. 6870, pp. 429–433, 2002.

[7] A. L. Yuille, H. H. Bulthoff, D. Kersten, and P. Mamassian, “Perception as Bayesian Inference,” *Annual Review of Psychology*, vol. 55, pp. 271–304, 1996.

[8] S. Deneve, “Bayesian spiking neurons I: inference.,” *Neural computation*, vol. 20, no. 1, pp. 91–117, Jan. 2008.

[9] B. R. Gaines, “Stochastic Computing Systems,” *Advances in information systems science. Springer US*, 1969.

[10] C. S. Thakur, S. Afshar, R. M. Wang, T. J. Hamilton, J. Tapson, and A. van Schaik, “Bayesian Estimation and Inference using Stochastic Hardware,” *Frontiers in Neuroscience*, vol. 10, no. March, pp. 1–28, 2015.