UNIVERSITY OF LEEDS

Proceedings Paper:
Yu, Z, Huang, T and Liu, JK orcid.org/0000-0002-5391-7213 (2018) Implementation of Bayesian Inference In Distributed Neural Networks. In: 2018 26th Euromicro International Conference on Parallel, Distributed and Network-based Processing (PDP). 2018 26th Euromicro International Conference on Parallel, Distributed and Network-based Processing (PDP), 21-23 Mar 2018, Cambridge, UK. IEEE , pp. 666-673. ISBN 978-1-5386-4976-3

https://doi.org/10.1109/pdp2018.2018.00111

Implementation of Bayesian inference in distributed neural networks

Zhaofei Yu†‡, Tiejun Hang†, Jian K. Liu ‡
†School of Electronics Engineering and Computer Science, Peking University, Beijing, China
‡Institute for Theoretical Computer Science, Graz University of Technology, Graz, Austria
Email: yzf714@126.com, tjhuang@pku.edu.cn, liu@igi,tugraz.at

***Abstract*—Numerous neuroscience experiments have suggested that the cognitive process of human brain is realized as probability reasoning and further modeled as Bayesian inference. It is still unclear how Bayesian inference could be implemented by neural underpinnings in the brain. Here we present a novel Bayesian inference algorithm based on importance sampling. By distributed sampling through a deep tree structure with simple and stackable basic motifs for any given neural circuit, one can perform local inference while guaranteeing the accuracy of global inference. We show that these task-independent motifs can be used in parallel for fast inference without iteration and scale-limitation. Furthermore, experimental simulations with a small-scale neural network demonstrate that our distributed sampling-based algorithm, consisting with our theoretical analysis, can approximate Bayesian inference. Taken all together, we provide a proof-of-principle to use distributed neural networks to implement Bayesian inference, which gives a road-map for large-scale Bayesian network implementation based on spiking neural networks with computer hardwares, including neuromorphic chips.**

***Keywords*-Bayesian inference; distributed neural network; importance sampling; neural implementation**

I. Introduction

Our brain can represent and process information with uncertainty [1]. It has been suggested by numerous physiological and psychological experiments that the cognitive behavior is a process of probabilistic reasoning based on Bayesian inference [2], [3] . From the macroscopic viewpoint, Bayesian model has been successfully used to explain these cognitive behaviors [3], [4]. However, from the microscopic perspective, it remains unclear how Bayesian inference is implemented in neuronal circuits.

According to recent studies, many researchers have devoted to proposing different neural circuits to represent and implement inference of Bayesian models. These neural circuits are mostly based on the inference algorithm of belief propagation (BP). Rao [5], [6] derived the inference equation of hidden Markov models (HMMs) with BP and demonstrated that the differential physical equation of recurrent neural circuits is consistent with the inference equation of HMMs, where a sum-logs is used to approximate a log-sum. Beck and Pouget [7] went a future step to solve the approximation problem and came up with a precise equivalence relation. Similarly, Ott et al. [8] and Yu et al. [9] built the relationship between inference equation of Markov random fields and the dynamics of recurrent neural networks with BP. The above works based on equivalence proof are only appropriate for small-scale Bayesian inference. Another important approach is to implement BP with neural circuits directly. George [10] and Hawkins rewrote BP of tree-structured Bayesian model with five equations and designed five basic neural circuits to implement these equations respectively. Steimer et al. [11] and Litvak et al. [12] generalized the result to inference of graphical models. These neural circuits are very complex and require each group of neurons to realize distinct and complex functions. Moreover, these inference methods need multiple iterations with slow speed.

In summary, these previous studies focus on how neural network implements inference for the simple Bayesian model with a small number of variables. In addition, they are difficult to be generalized as they are task-specific [13]. Therefore it is necessary to propose a new algorithm which could perform rapid inference for large-scale Bayesian model and be implemented by simple neural circuits efficiently. Here we propose a distributed sampling-based algorithm for Bayesian inference that can be easily implemented in neural networks. In particular, our algorithm takes advantage of the four principles of neural system: scalability with a large number of neurons; hierarchy with multiple layers; locality with computation done within a relatively small group of neurons; parallelizability with computations distributed simultaneously.

In short, our main idea of the sampling-based inference is to perform sampling on a deep tree-structured model. With the benefit of tree structure, the global inference problem is converted to the local inference problem. As a result, we are able to design sampling-based inference algorithm for local inference problem while guaranteeing the accuracy of global inference. On the local level, we introduce importance sampling to perform inference, which utilizes massive number of samples to compute in parallel and calculates without iteration. With this strategy of trading space for time, inference would be implemented rapidly. We theoretically prove that Bayesian inference can be approximated in such a hierarchical structure with a distributed fashion. Experimental simulations of multi-cue information demonstrate that the proposed algorithm can achieve the adequate accuracy for Bayesian inference.

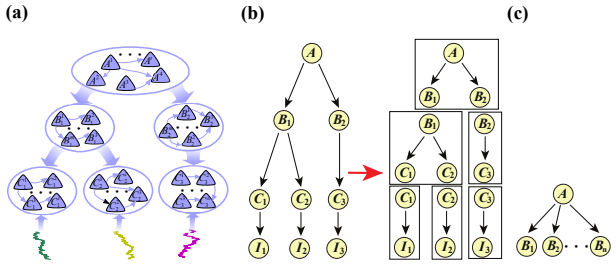

Figure 1. Neural network model represented by Bayesian network. (a) An illustration of typical neural network model with computation done by hierarchical and localized structures. Here is a three-layer (represented by A, B, C for each layer) network with input evidence in three difference sources (indicated by green, yellow, purple, respectively). Each triangle represents a neuron. Each circle represents a group of neurons for local computation (the Winner-Take-All circuit, for instance). Then, C_1^1 represents the neuron No. 1 of the group No. 1 in the first layer C. (b) Represented Bayesian network corresponding to the neural network in (a) with a tree-structure (left) can be decomposed into distributed basic network motifs (right). The basic network in each box is composed of a parent node and a group of children nodes. (c) The basic network motif with one parent node A and a group of children nodes $B_i, (i = 1, \ldots, n)$. In (b) and (c), the evidence are represented by $I_i, (i = 1, 2, 3)$, and each symbol (A, B, C), corresponding to the blue circle of neurons in (a), represents a group of neurons for local computation.

The rest of this paper is organized as follows. We first introduce the sampling-based inference algorithm in section II, and then give some theoretical analysis of inference in section III. We show the experimental results in section IV and conclude in section V.

II. Bayesian inference with importance sampling

In this section, we consider how to realize inference for Bayesian models. In particular, we consider a network of Bayesian model with tree-structure that has been studied intensively [14]. Several inference methods, such as belief propagation (BP) [14] and Markov chain Monte Carlo (MCMC) [15], can get accurate results with the benefit of tree structure. In addition, the tree-structured Bayesian models could represent other non-tree structured Bayesian models since they could be converted to tree-structured Bayesian models by combining some variables together [14].

A. Decomposition of global inference to local inference

Inference of a tree-structured Bayesian model is to infer the state of the root node according to the states of leaf nodes by calculating posterior probability and the maximum of a posterior probability. To be specific, considering a generic neural network typically used for modeling in Fig. 1a, one can represent it with a tree-structured Bayesian network as in Fig. 1b, the root node is A and the leaf nodes are I_1, I_2 and I_3. Supposing that we have known the prior probability $P(A)$ and conditional probabilities $P(B_1|A)$, $P(B_2|A)$, $P(C_1|B_1)$, $P(C_2|B_1)$, $P(C_3|B_2)$, $P(I_1|C_1)$, $P(I_2|C_2)$, $P(I_3|C_3)$. Since the states of the leaf nodes are also known, one can express the inference problem as follows:

- computing posterior probability $P(A|I_1, I_2, I_3)$
- maximum a posterior (MAP) estimation $\arg\max_A P(A|I_1, I_2, I_3)$.

As seen in Fig. 1b, the beliefs propagate from bottom to top when we infer the state of the root node. One can decompose the whole network into a set of simple subnetworks. Each subnetwork is able to receive beliefs from the children nodes and pass its belief to the parent nodes. Note that these simple subnetworks share similar structures and consist of a set of basic network motifs as in each box of Fig. 1b. In the end, this is the *only* type of most basic network motif with *one* parent node and a group of children nodes (shown in Fig. 1c). One only need to design a suitable algorithm to perform inference for this most basic network motif. Then the implementation of all basic motifs can be combined to perform inference of the whole network problem from bottom to top.

B. Inference with importance sampling

Importance sampling is a method to calculate the probability by sampling from a simple distribution (a distribution from which the samples are easy to be generated, e.g. in terms of a Gaussian distribution or a uniform distribution) rather than the actual distribution to be computed [16]. Shi and Griffiths [17] used importance sampling to calculate the conditional expectation of some function over a discrete random variable x given y:

$$\begin{aligned} E(f(x)|y) &= \sum_x f(x) P(x|y) = \frac{E(f(x)P(y|x))_{P(x)}}{E(P(y|x))_{P(x)}} \\ &\approx \sum_{x^i} f(x^i) \frac{P(y|x^i)}{\sum_{x^i} P(y|x^i)}, \qquad x^i \sim P(x), \end{aligned} \tag{1}$$

where $x^i \sim P(x)$ shows that x^i is drawn from the distribution $P(x)$. Note that x can be seen as the parent node of y. Equation (1) converts conditional expectation to the weighted combination of normalized conditional probabilities with samples drawn from the prior probability, which means we can calculate the expectation of a parent node with samples of its children nodes.

Equation (1) can be generalized to perform inference of the basic motif in Fig. 1c. The inference problem is to calculate posterior probability $P(A|I_1, I_2, ..., I_n)$, where $I_1, I_2, ..., I_n$ represent evidence variables of $B_1, B_2, ..., B_n$ respectively. By using importance sampling, this problem can be converted to:

$$\begin{aligned} &P(A|I_1, I_2, ..., I_n) \\ &= \sum_{B_1,...,B_n} P(A|B_1, ..., B_n) P(B_1, ...B_n|I_1, ...I_n) \\ &\approx \sum_i P(A|B_1^i, B_2^i, ..., B_n^i) \frac{P(I_1, I_2, ...I_n|B_1^i, B_2^i, ..., B_n^i)}{\sum_i P(I_1, I_2, ...I_n|B_1^i, B_2^i, ..., B_n^i)} \\ &= \sum_i P(A|B_1^i, B_2^i, ..., B_n^i) \frac{P(I_1|B_1^i)P(I_2|B_2^i)...P(I_n|B_n^i)}{\sum_i P(I_1|B_1^i)P(I_2|B_2^i)...P(I_n|B_n^i)} \\ &\qquad B_1^i, B_2^i, ..., B_n^i \sim P(B_1, B_2, ..., B_n). \end{aligned} \tag{2}$$

Equation (2) can be used for further inference when A is a child node of other nodes. Therefore, this is the most

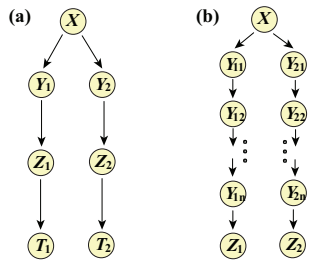

Figure 2. Basic models for justification of conditional independence assumptions. (a) A simple Bayesian network for Theorem 1 and Set 1 assumptions, where X expresses a parent node or ancestor nodes of Y_1, Y_2. (b) A multi-hierarchy Bayesian network for Theorem 2 and Set 2 assumptions.

fundamental building block of computation for any size of network with multiple groups of evidence and layers to be distributed in parallelism.

Note that the number of children nodes n in the generic motifs is arbitrary. Therefore, by using Equation (2), we can perform inference for any tree-structured Bayesian model by decomposing it into a set of basic motifs as in Fig. 1b (right). Here we illustrate the process with the example of calculation for the model in Fig. 1b (left), we have the posterior probability calculated by Equation (3) (next page):

Here $I(A = a_1)$ is an indicative function, which equals to 1 when $A = a_1$ holds and 0 elsewise. $C_1^i, C_2^i \sim P(C_1, C_2)$, $C_3^j \sim P(C_3)$, $B_1^k, B_2^k \sim P(B_1, B_2)$, $A^l \sim P(A)$. In this example, the inference Equation (3) is based on a series of conditional independence assumptions:

Set 1: $B_1 \perp C_3 | C_1, C_2$, $B_1 \perp B_2 | C_3$, $B_1 \perp C_3^j | C_1^i, C_2^i$, and $B_1 \perp B_2 | C_3^j$,

Set 2: $C_1, C_2 \perp I_3 | I_1, I_2$ and $C_1, C_2 \perp C_3 | I_3$.

We will give the theoretical analysis of these conditional independence assumptions in the next section.

Correspondingly, MAP estimation $\arg\max_A P(A|I_1, I_2, I_3)$ is to find the state that maximize the posterior probability, which can be calculated easily given the posterior probability $P(A|I_1, I_2, I_3)$.

C. Calculation of prior probabilities based on importance sampling

The inference algorithm above should meet the requirement that the samples are drawn from the prior probabilities. However, we don't know all the prior probabilities. Considering Equation (3), there are four prior probabilities which should be known in advance, including $P(C_1, C_2)$, $P(C_3)$, $P(B_1, B_2)$ and $P(A)$, but we only know $P(A)$ and some conditional probabilities. An algorithm should be designed to calculate all the prior probabilities. Interestingly, importance sampling could also be used to calculate the prior probabilities:

$$P(B_1, B_2) = \sum_A P(A, B_1, B_2) \approx \frac{1}{M} \sum_{i=1}^{M} P\left(B_1, B_2 | A^i\right). \quad (4)$$

Here A^i is drawn from the distribution $P(A)$. Then the posterior probabilities of $P(C_1, C_2)$ and $P(C_3)$ could be calculated based on $P(B_1, B_2)$. For example, $P(C_3)$ is calculated by:

$$P(C_3) = \sum_{B_2} P(B_2, C_3) \approx \frac{1}{M} \sum_{i=1}^{M} P\left(C_3 | B_2^i\right). \quad (5)$$

All together, our proposed inference algorithm based on importance sampling could perform fast inference for tree-structured Bayesian model. The strategy of local inference is comparable to the idea of local computation done by some neural circuit motifs, such as cortical minicolumn in different sensory modalities in neuronal system.

III. Theoretical analysis of conditional independence assumptions

Our proposed inference algorithm above includes a series of conditional independence assumptions. Now we will prove that they do not effect the inference accuracy and the results will converge to the exact solution as the sample size and the network layers go to infinity. The following theorems resolve these two sets of assumptions in Equation (3) respectively.

Theorem 1. *Considering a Bayesian network as in Fig. 2a, we define that:*

$$f_1(Y_1, Y_2) = \sum_{Z_1, Z_2} P(Y_1, Y_2 | Z_1, Z_2) P(Z_1 | T_1) P(Z_2 | T_2),$$

$$f_2(Y_1, Y_2) = \sum_{i=1}^{M} \sum_{j=1}^{N} P\left(Y_1, Y_2 | Z_1^i, Z_2^j\right) \frac{P\left(T_1 | Z_1^i\right)}{\sum_{i=1}^{M} P\left(T_1 | Z_1^i\right)} \frac{P\left(T_2 | Z_2^j\right)}{\sum_{j=1}^{N} P\left(T_2 | Z_2^j\right)},$$

$$Z_1^i \sim P(Z_1), Z_2^j \sim P(Z_2),$$

then for arbitrary small number ε,

$$\lim_{\substack{M \to \infty \\ N \to \infty}} P\left(|f_2(Y_1, Y_2) - f_1(Y_1, Y_2)| < \varepsilon\right) = 1. \quad (6)$$

The proof of Theorem 1 is provided in Appendix A. Theorem 1 means that $f_2(Y_1, Y_2)$ is an estimator of $f_1(Y_1, Y_2)$ and converges to $f_1(Y_1, Y_2)$ with probability 1 when M and N tend to infinite. Based on this theorem, it is easy to show that assumptions Set 1 will not affect the accuracy of our algorithm. Now note that the inference process in Equation (3) can be expressed as a series of four steps g_1 to g_4 below:

$$g_1 = \sum_A I(A = a_1) \sum_{B_1, B_2} P(A | B_1, B_2) \sum_{C_1, C_2, C_3} \{ \quad (7)$$
$$P(C_1, C_2 | I_1, I_2) P(C_3 | I_3) P(B_1, B_2 | C_1, C_2, C_3) \},$$

$$
\begin{aligned}
& P\left(A=a_1|I_1,I_2,I_3\right) \qquad (3)\\
=& \sum_{A,B_1,B_2,C_1,C_2,C_3} I\left(A=a_1\right) P\left(A,B_1,B_2,C_1,C_2,C_3|I_1,I_2,I_3\right)\\
=& \sum_{A,B_1,B_2,C_1,C_2,C_3} I\left(A=a_1\right) P\left(C_1,C_2,C_3|I_1,I_2,I_3\right) P\left(B_1,B_2|C_1,C_2,C_3\right) P\left(A|B_1,B_2\right)\\
=& \sum_{A,B_1,B_2,C_1,C_2,C_3} I\left(A=a_1\right) P\left(C_1,C_2|I_1,I_2,I_3\right) P\left(C_3|C_1,C_2,I_3\right) P\left(B_1,B_2|C_1,C_2,C_3\right) P\left(A|B_1,B_2\right)\\
\approx& \sum_{A,B_1,B_2,C_1,C_2,C_3} I\left(A=a_1\right) P\left(C_1,C_2|I_1,I_2\right) P\left(C_3|I_3\right) P\left(B_1,B_2|C_1,C_2,C_3\right) P\left(A|B_1,B_2\right)\\
\approx& \sum_{A,B_1,B_2,C_1,C_2,C_3} I\left(A=a_1\right) P\left(C_1,C_2|I_1,I_2\right) P\left(C_3|I_3\right) P\left(B_1|C_1,C_2\right) P\left(B_2|C_3\right) P\left(A|B_1,B_2\right)\\
\approx& \sum_{A,B_1,B_2} I\left(A=a_1\right) P\left(A|B_1,B_2\right) \left(\sum_i P\left(B_1|C_1^i,C_2^i\right) \frac{P\left(I_1,I_2|C_1^i,C_2^i\right)}{\sum\limits_i P\left(I_1,I_2|C_1^i,C_2^i\right)}\right) \left(\sum_j P\left(B_2|C_3^j\right) \frac{P\left(I_3|C_3^j\right)}{\sum\limits_j P\left(I_3|C_3^j\right)}\right)\\
\approx& \sum_{A,B_1,B_2} I\left(A=a_1\right) P\left(A|B_1,B_2\right) \sum_i \sum_j P\left(B_1,B_2|C_1^i,C_2^i,C_3^j\right) \frac{P\left(I_1,I_2|C_1^i,C_2^i\right)}{\sum\limits_i P\left(I_1,I_2|C_1^i,C_2^i\right)} \frac{P\left(I_3|C_3^j\right)}{\sum\limits_j P\left(I_3|C_3^j\right)}\\
\approx& \sum_{A,i,j} I\left(A=a_1\right) \sum_k P\left(A|B_1^k,B_2^k\right) \frac{P\left(C_1^i,C_2^i,C_3^j|B_1^k,B_2^k\right)}{\sum\limits_k P\left(C_1^i,C_2^i,C_3^j|B_1^k,B_2^k\right)} \frac{P\left(I_1,I_2|C_1^i,C_2^i\right)}{\sum\limits_i P\left(I_1,I_2|C_1^i,C_2^i\right)} \frac{P\left(I_3|C_3^j\right)}{\sum\limits_j P\left(I_3|C_3^j\right)}\\
\approx& \sum_l I\left(A^l=a_1\right) \sum_k \frac{P\left(B_1^k,B_2^k|A^l\right)}{\sum\limits_l P\left(B_1^k,B_2^k|A^l\right)} \sum_{i,j} \frac{P\left(C_1^i,C_2^i,C_3^j|B_1^k,B_2^k\right)}{\sum\limits_k P\left(C_1^i,C_2^i,C_3^j|B_1^k,B_2^k\right)} \frac{P\left(I_1,I_2|C_1^i,C_2^i\right)}{\sum\limits_i P\left(I_1,I_2|C_1^i,C_2^i\right)} \frac{P\left(I_3|C_3^j\right)}{\sum\limits_j P\left(I_3|C_3^j\right)}.
\end{aligned}
$$

$$
\begin{aligned}
g_2 = \sum_A I\left(A=a_1\right) \sum_{B_1,B_2} P\left(A|B_1,B_2\right) \sum_{C_1,C_2,C_3} \{ \qquad (8)\\
P\left(C_1,C_2|I_1,I_2\right) P\left(C_3|I_3\right) P\left(B_1|C_1,C_2\right) P\left(B_2|C_3\right) \},
\end{aligned}
$$

$$
\begin{aligned}
g_3 = \sum_A I\left(A=a_1\right) \sum_{B_1,B_2} \Bigg\{ P\left(A|B_1,B_2\right) \left(\sum_i P\left(B_1|C_1^i,C_2^i\right) \right. \qquad (9)\\
\left. \frac{P\left(I_1,I_2|C_1^i,C_2^i\right)}{\sum\limits_i P\left(I_1,I_2|C_1^i,C_2^i\right)}\right) \left(\sum_j P\left(B_2|C_3^j\right) \frac{P\left(I_3|C_3^j\right)}{\sum\limits_j P\left(I_3|C_3^j\right)}\right) \Bigg\}\\
C_1^i, C_2^i \sim P\left(C_1,C_2\right) \quad C_3^j \sim P\left(C_3\right),
\end{aligned}
$$

$$
\begin{aligned}
g_4 = \sum_A I\left(A=a_1\right) \sum_{B_1,B_2} P\left(A|B_1,B_2\right) \sum_i \sum_j \{ \qquad (10)\\
P\left(B_1,B_2|C_1^i,C_2^i,C_3^j\right) \frac{P\left(I_1,I_2|C_1^i,C_2^i\right)}{\sum\limits_i P\left(I_1,I_2|C_1^i,C_2^i\right)} \frac{P\left(I_3|C_3^j\right)}{\sum\limits_j P\left(I_3|C_3^j\right)} \Bigg\}\\
C_1^i, C_2^i \sim P\left(C_1,C_2\right) \quad C_3^j \sim P\left(C_3\right).
\end{aligned}
$$

The transformation from Equation (7) to Equation (8) includes the conditional independence assumptions $B_1 \perp C_3 | C_1, C_2$, $B_1 \perp B_2 | C_3$. Equation (9) is the importance sampling result of Equation (8). From Equation (9) to Equation (10), we use the conditional independence assumptions $B_1 \perp C_3^j | C_1^i, C_2^i$, $B_1 \perp B_2 | C_3^j$. With theorem 1, it is easy to show that for arbitrary small number ε, $\lim\limits_{\substack{M\to\infty \\ N\to\infty}} P\ \left(|g_4 - g_1| < \varepsilon\right) = 1$, where M and N are the sample sizes of C_1^i, C_2^i and C_3^j respectively.

Therefore, this result shows that assumptions Set 1 have no influence on the accuracy of our algorithm. We treat Equation (10) as a generalized importance sampling result of Equation (7). In a biological neural system this inference process can be implemented by neurons with simple operations. This result is universal for different models as long as it has a structure as in Fig. 2a.

Theorem 2. *Considering a Bayesian network as Fig. 2b shows, the prior probabilities $P\left(X\right)$ and conditional probabilities $P\left(Z_t|Y_{t,n}\right)$ are random for $t = 1, 2$. Similarly, the conditional probabilities $P\left(Y_{t,1}|X\right)$ and $P\left(Y_{t,i+1}|Y_{t,i}\right)$ are random and non-zero for $i = 1, 2, ..., n-1$ and $t = 1, 2$. Then we conclude that $Z_1 \perp Z_2$ when n tends to infinite.*

The proof is provided in Appendix B. This theorem states that the dependence between Z_1 and Z_2 decreases as the hierarchy increases and will converge to zero when

the hierarchy tends to infinite. In practice, we found that variables Z_1 and Z_2 are already approximately independently when the hierarchy has two layers in our numerical experiments. Assumptions Set 2 can be justified by Theorem 2. It is easy to show that the variables C_1, C_2 and C_3 are approximately independent, as a result, $P(C_1, C_2, C_3) = P(C_1, C_2) P(C_3)$. Then we can get:

$$\begin{aligned} & P(C_1, C_2 | I_1, I_2, I_3) \\ &= \frac{\sum_{C_3} P(C_1, C_2, C_3, I_1, I_2, I_3)}{\sum_{C_1, C_2, C_3} P(C_1, C_2, C_3, I_1, I_2, I_3)} \\ &= \frac{\sum_{C_3} P(C_1, C_2) P(C_3) P(I_1, I_2 | C_1, C_2) P(I_3 | C_3)}{\sum_{C_1, C_2} P(C_1, C_2) P(I_1, I_2 | C_1, C_2) \sum_{C_3} P(C_3) P(I_3 | C_3)} \\ &= \frac{P(I_1, I_2, C_1, C_2) P(I_3)}{P(I_1, I_2) P(I_3)} = P(C_1, C_2 | I_1, I_2), \end{aligned} \tag{11}$$

$$\begin{aligned} P(C_3 | C_1, C_2, I_3) &= \frac{P(C_1, C_2, C_3, I_3)}{\sum_{C_3} P(C_1, C_2, C_3, I_3)} \\ &= \frac{P(C_1, C_2) P(C_3) P(I_3 | C_3)}{\sum_{C_3} P(C_1, C_2) P(C_3) P(I_3 | C_3)} = P(C_3 | I_3), \end{aligned} \tag{12}$$

which proves $C_1, C_2 \perp I_3 | I_1, I_2$, $C_1, C_2 \perp C_3 | I_3$ as in Set 2 assumptions. In the perspective of probabilistic graphical models, C_1, C_2 and C_3 are not independent. However, in a biological neural system, this independence can be hold approximately since there are multiple layers organized in a hierarchy fashion. For instance, the ventral visual pathway starts from the retina to the visual cortex and reaches inferior temporal cortex [18]. An intuitive understanding is that if the neurons representing C_1, C_2 affect the neurons representing C_3, it should pass belief to C_3 through A. As the path becomes longer enough, the effects will become smaller and close to zero.

With these two theorems together, we have proved that all the conditional independence assumptions raised in our algorithm do not affect the inference accuracy.

IV. Simulations

We test the accuracy of our proposed algorithm by using a classical problem of the sensory integration of multi-cue information. Certainly one can test it with more complex cognitive tasks with a large scale of hierarchical Bayesian model, which is beyond the current study.

Human brain could receive cues from multiple sensory modalities and then integrate them in an optimal way, which is called multi-cue integration. To be specific, when we hear a sound from an object, look at it and touch it simultaneously, we receive auditory, visual and somatosensory information. We consider a 3-cue integration problem, which could be modeled by a two-layer Bayesian network (shown in Fig. 3a). Here S represents the location of the stimulus, S_H, S_V and S_A denote visual, haptic, and auditory cues respectively. Supposing that $P(S)$ is a uniform distribution,

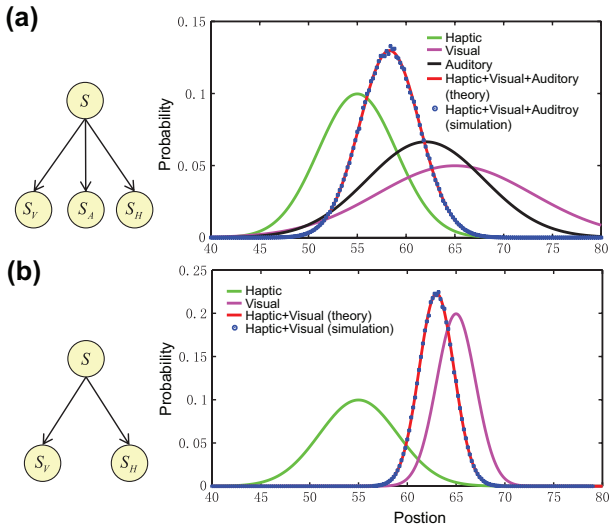

Figure 3. Simulations of multi-cue integration. (a) A two-layer Bayesian model for haptic-visual-auditory integration (left). (right) The result of our method compared to the theoretical value. Theoretical result indicated in red, and simulation indicated in blue. $\sigma^2_{S_H} = 64$, $\sigma^2_{S_V} = 16$ and $\sigma^2_{S_A} = 36$. Each point is averaged over 10 trials of different results of sampling. (b) Similar to (a) but for haptic-visual integration. $\sigma^2_{S_H} = \sigma^2_{S_V} = 16$.

$P(S_H|S)$, $P(S_V|S)$ and $P(S_A|S)$ are three different Gaussian distributions with the same mean value S and different variances $\sigma^2_{S_H}$, $\sigma^2_{S_V}$ and $\sigma^2_{S_A}$. Then we can use importance sampling to infer the state of S given S_H, S_V and S_A:

$$\begin{aligned} P(S = s | S_H, S_V, S_A) &= \sum_S I(S = s) P(S | S_H, S_V, S_A) \\ &= \sum_i I(S_i = s) \frac{P(S_H, S_V, S_A | S_i)}{\sum_i P(S_H, S_V, S_A | S_i)} \quad S_i \sim P(S). \end{aligned} \tag{13}$$

The neural circuit to implement inference for this problem is based on probabilistic population coding and two plausible neural operations: normalization and linear combinations. There are 1000 Poisson spiking neurons which encode stimuli. The distributions of these Poisson spiking neurons are determined by prior probabilities $P(S)$. The tuning curve of the neuron i is proportional to the conditional probability $P(S_H, S_V, S_A | S_i)$. Then the inhibitory neurons are used to get normalization. If we use y_i to express the individual firing rate of Poisson spiking neuron i and Y to express the total firing rate as $Y = \sum_i y_i$, we can get $E(y_i/Y = n) = \frac{P(S_H, S_V, S_A | S_i)}{\sum_i P(S_H, S_V, S_A | S_i)}$. The normalized results are linearly combined with their synaptic weights $I(S_i = s)$ to get the inference result $P(S = s | S_H, S_V, S_A)$.

Simulation results are shown in Fig. 3a, where the modeling results obtained by our proposed method with important sampling fit the theoretical value very well. A previous study [17] implemented Bayesian inference with importance sampling for 2-cue integration. Here we illustrate the case of 2-cue integration for the completeness as in Fig. 3b.

V. Conclusion and discussion

Theoretically, it is important to understand how the neural network performs inference in a Bayesian fashion. In this study, we proposed the sampling-based inference algorithm, which is a distributed algorithm for large-scale Bayesian model. We showed by theoretical analysis and simulations that our method can generate the accurate inference.

A. Comparison to previous work

Shi and Griffiths [17] have shown that the inference of chain Bayesian network with importance sampling can be implemented by neural networks. Our work is an extension to more general Bayesian network. Besides, here we proved the convergence of the sampling-based inference method, which is not discussed in [17].

For any non-tree structured feedford network, one can transfer it into a more general tree structured network [3] by combining those relevant variables together at the cost of greater state space [14]. Thus more neurons are needed to express all the states so that one can speed up inference by avoiding temporal iterations with more neurons sampled over space but in a parallel and local fashion.

B. Distributed computation

Distributed Bayesian inference has become a rich research direction [19]. In addition, it has been suggested that human collective intelligence follows distributed Bayesian inference [20]. With the great advancements of hardware devices, including neuromorphic chips in recent years, we expect that our method can be implemented in these hardwares. The hardware devices also provide the base for large-scale distributed Bayesian inference, which is the main feature of our algorithm.

Bayesian computations have been implemented on hardwares according to specific tasks [21]. However, most of the previous studies are to split the data into small parts, then perform the inference for each part independently, and combine the results in the end [22]. Such an approach violates the principle that each separate part/area in the brain should exchange information with the others. The inference algorithm we proposed takes advantage of this principle, specifically, each part of the neural network can exchange information with neighboring networks (parent and children networks), which may shed new light on neural plausible implementation of distributed Bayesian inference.

There are different representations of distribution dependent on the context. For instance, in terms of sensory inputs in our brain, different resources, such as visual, auditory, haptic inputs, and etc., are processed individually by the corresponding sensory organs. Even in each sensory organ, different features are processed by different types of neurons. In the retina, there are many types of retinal ganglion cells to compute different visual features, such as contrast, spatial and temporal frequencies, speed, orientation, direction, etc. [23]. In neuromorphic engineering, one could represent these different features by some hard-coded circuits, for example, a circuit of event pixels based on the dynamics visual sensor of silicon retina for objection motion [24]. Such feature specific circuits could be furthermore distributed in hardware to implement complex tasks.

Here we only conducted some simple experiments of multi-cue integration. Although most of the current neuroscience experiments are conducted for relatively simple cognition behaviors, some more complex tasks have been proposed, for example hierarchical decision-making task [25]. We are making some efforts in this direction in larger-scale of simulations and hardware implementations.

Appendix A.
Proof of Theorem 1

Proof of Theorem 1. We rewrite $f_2(Y_1, Y_2)$ as

$$
\begin{aligned}
& f_2(Y_1, Y_2) \qquad (14)\\
&= \sum_{i=1}^{M}\sum_{j=1}^{N} P\left(Y_1, Y_2 | Z_1^i, Z_2^j\right) \frac{P\left(T_1|Z_1^i\right)}{\sum_{i=1}^{M} P\left(T_1|Z_1^i\right)} \frac{P\left(T_2|Z_2^j\right)}{\sum_{j=1}^{N} P\left(T_2|Z_2^j\right)} \\
&\qquad\qquad Z_1^i \sim P(Z_1), \;\; Z_2^j \sim P(Z_2) \\
&= \frac{\frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N} P\left(Y_1, Y_2 | Z_1^i, Z_2^j\right) P\left(T_1|Z_1^i\right) P\left(T_2|Z_2^j\right)}{\frac{1}{MN}\sum_{k=1}^{M}\sum_{l=1}^{M} P\left(T_1|Z_1^k\right) P\left(T_2|Z_2^l\right)}, \\
&\qquad Z_1^i \sim P(Z_1), \;\; Z_2^j \sim P(Z_2), \;\; Z_1^k \sim P(Z_1) \;\; Z_2^l \sim P(Z_2)
\end{aligned}
$$

The expectation and variance take the form

$$
\begin{aligned}
& E\left(P\left(Y_1, Y_2 | Z_1^i, Z_2^j\right) P\left(T_1|Z_1^i\right) P\left(T_2|Z_2^j\right)\right) \qquad (15)\\
&= \sum_{Z_1^i}\sum_{Z_2^j} \left\{P\left(Y_1, Y_2 | Z_1^i, Z_2^j\right) P\left(T_1|Z_1^i\right) P\left(T_2|Z_2^j\right) P\left(Z_1^i\right) P\left(Z_2^j\right)\right\} \\
&= \sum_{Z_1}\sum_{Z_2} P(Y_1, Y_2 | Z_1, Z_2) P(T_1, Z_1) P(T_2, Z_2) \\
&= f_1(Y_1, Y_2) P(T_1) P(T_2),
\end{aligned}
$$

$$
\begin{aligned}
& E\left(P\left(T_1|Z_1^k\right) P\left(T_2|Z_2^l\right)\right) \qquad (16)\\
&= \sum_{Z_1^k}\sum_{Z_2^l} P\left(T_1|Z_1^k\right) P\left(T_2|Z_2^l\right) P\left(Z_1^k\right) P\left(Z_2^l\right) = P(T_1) P(T_2),
\end{aligned}
$$

$$
\begin{aligned}
& Var\left(P\left(Y_1, Y_2 | Z_1^i, Z_2^j\right) P\left(T_1|Z_1^i\right) P\left(T_2|Z_2^j\right)\right) \qquad (17)\\
&= \sum_{Z_1}\sum_{Z_2} \Big\{P(Y_1, Y_2|Z_1, Z_2)^2 P(T_1|Z_1)^2 P(T_2|Z_2)^2 \\
&\quad \cdot P(Z_1)^2 P(Z_2)^2\Big\} - f_1(Y_1, Y_2)^2 P(T_1)^2 P(T_2)^2,
\end{aligned}
$$

$$
\begin{aligned}
& Var\left(P\left(T_1|Z_1^k\right) P\left(T_2|Z_2^l\right)\right) \qquad (18)\\
&= \sum_{Z_1}\sum_{Z_2} P(T_1|Z_1)^2 P(T_2|Z_2)^2 P(Z_1) P(Z_2) - P(T_1)^2 P(T_2)^2.
\end{aligned}
$$

Since $f_1(Y_1, Y_2) P(T_1) P(T_2) / P(T_1) P(T_2) = f_1(Y_1, Y_2)$, it is easy to show that for arbitrary small number ε,

$$\lim_{\substack{M\to\infty \\ N\to\infty}} P\left(|f_2(Y_1, Y_2) - f_1(Y_1, Y_2)| < \varepsilon\right) = 1.$$

APPENDIX B.
PROOF OF THEOREM 2

Lemma 1. *Supposing that $A_1, A_2, ..., A_n$ is randomly generated matrix with the equality that $row(A_i) = col(A_{i+1})$ for $i = 1, 2, ..., n$. The arbitrary element in $A_1, A_2, ..., A_n$ is in $[\varepsilon, 1-\varepsilon]$, where ε is a small number. Besides, the sum of arbitrary row of arbitrary matrix $A_1, A_2, ..., A_n$ is 1. We define that: $C_k = \left(\prod_{i=1}^{k} A_i^T\right)^T$, we can conclude that all elements in a special col of C_k will tend to a same value when k tends to infinity. Proof of theorem 2* It is easy to get that $C_i = A_i C_{i-1}$ if $i \geq 2$ and $C_i = A_i$ if $i = 1$. Besides, $col(C_i) = col(A_1)$, $row(C_i) = row(A_i)$. Supposing that $A_i = \begin{bmatrix} a_{i,1,1} & a_{i,1,2} & \cdots & a_{i,1,n(i)} \\ a_{i,2,1} & a_{i,2,2} & \cdots & a_{i,2,n(i)} \\ . & . & . & . \\ . & . & . & . \\ a_{i,m(i),1} & a_{i,m(i),2} & \cdots & a_{i,m(i),n(i)} \end{bmatrix}$, $C_i = \begin{bmatrix} c_{i,1,1} & c_{i,1,2} & \cdots & c_{i,1,n(1)} \\ c_{i,2,1} & c_{i,2,2} & \cdots & c_{i,2,n(1)} \\ . & . & . & . \\ . & . & . & . \\ c_{i,m(i),1} & c_{i,m(i),2} & \cdots & c_{i,m(i),n(1)} \end{bmatrix}$, where $m(i)$ and $n(i)$ are function of i which show the row and col of the matrix A_i. We use $\widehat{c}_{i,j}$ to express the vector of all the elements in col j of matrix C_i. Then $\max(\widehat{c}_{i,j})$ represents the maximum element in col j of matrix C_i and $\min(\widehat{c}_{i,j})$ represents the minimum element in col j of matrix C_i. Now for arbitrary $c_{i+1,s,t}$, where $s \in (1, 2, ..., m(i+1)), t \in (1, 2, ..., n(1))$, we can get:

$$c_{i+1,s,t} = a_{i+1,s,1}c_{i,1,t} + a_{i+1,s,2}c_{i,2,t} + ... + a_{i+1,s,n(i+1)}c_{i,m(i+1),t}. \tag{19}$$

As $\sum_{j=1}^{n(i+1)} a_{i+1,s,j} = 1$, the equation above can be treated as the weighted average of col t of matrix C_i. By using the condition that the arbitrary element of $A_1, A_2, ..., A_n$ is in $[\varepsilon, 1-\varepsilon]$, we can get that:

$$\begin{aligned}(1-\varepsilon)\min(\widehat{c}_{i,t}) + \varepsilon\max(\widehat{c}_{i,t}) &\leq c_{i+1,s,t} \\ &\leq \varepsilon\min(\widehat{c}_{i,t}) + (1-\varepsilon)\max(\widehat{c}_{i,t}),\end{aligned} \tag{20}$$

which is equivalent to

$$0 \leq \max(\widehat{c}_{i+1,t}) - \min(\widehat{c}_{i+1,t}) \leq (1-2\varepsilon)(\max(\widehat{c}_{i,t}) - \min(\widehat{c}_{i,t})). \tag{21}$$

The equation above can be rewritten as

$$0 \leq \max(\widehat{c}_{i+1,t}) - \min(\widehat{c}_{i+1,t}) \leq (1-2\varepsilon)^i(\max(\widehat{c}_{1,t}) - \min(\widehat{c}_{1,t})). \tag{22}$$

If we calculate the limit for both sides as i tends to infinite, we can get

$$\lim_{i\to\infty} (\max(\widehat{c}_{i+1,t}) - \min(\widehat{c}_{i+1,t})) = 0, \tag{23}$$

which means all elements in a special col of C_i will tend to a same value.

Proof of Theorem 2. Supposing that $U_{t,1}$ ($t = 1$ or 2) is a matrix with its element in row i and col j expressed as $u_{t,1,i,j}$ and $u_{t,1,i,j} = P(Y_{t,1} = Y_{t,1}(j)|X = X(i))$, where $Y_{t,1}(j)$ stands for jth element of variable $Y_{t,1}$ and $X(i)$ stands for ith element of variable X. Similarly, ($t = 1 or 2$ and $s = 1, 2, \ldots, n$) is a matrix with its element in row i and col j expressed as $U_{t,s}$ and $u_{t,s,i,j} = P(Y_{t,s} = Y_{t,s}(j)|Y_{t,s-1} = Y_{t,s-1}(i))$. Moreover, $U_{t,n+1}$ ($t = 1 or 2$) is a matrix with its element in row i and col j expressed as $u_{t,n+1,i,j}$ and $u_{t,n+1,i,j} = P(Z_{t,1} = Z_{t,1}(j)|Y_{t,n} = Y_{t,n}(i))$. Then

$$\begin{aligned}P(Z_1) &= \sum_X \sum_{Y_{1,1}} \sum_{Y_{1,2}} \cdots \sum_{Y_{1,n}} P(X) P(Y_{1,1}|X) P(Y_{1,2}|Y_{1,1}) \\ &\quad ... P(Y_{1,n}|Y_{1,n-1}) P(Z_1|Y_{1,n}) \\ &= \sum_X P(X) \sum_{Y_{1,1}} P(Y_{1,1}|X) \sum_{Y_{1,2}} P(Y_{1,2}|Y_{1,1}) \\ &\quad ... \sum_{Y_{1,n}} P(Y_{1,n}|Y_{1,n-1}) P(Z_1|Y_{1,n}) \\ &= \sum_X P(X) f(X, Z_1).\end{aligned} \tag{24}$$

Similarly,

$$\begin{aligned}P(Z_2) &= \sum_X \sum_{Y_{2,1}} \sum_{Y_{2,2}} \cdots \sum_{Y_{2,n}} P(X) P(Y_{2,1}|X) P(Y_{2,2}|Y_{2,1}) \\ &\quad ... P(Y_{2,n}|Y_{2,n-1}) P(Z_2|Y_{2,n}) \\ &= \sum_X P(X) g(X, Z_2),\end{aligned} \tag{25}$$

$$\begin{aligned}P(Z_1, Z_2) &= \sum_X \sum_{Y_{1,1}} \sum_{Y_{1,2}} \cdots \sum_{Y_{1,n}} \sum_{Y_{2,1}} \cdots \sum_{Y_{2,n}} \{P(X) P(Y_{1,1}|X) \\ &\quad ... P(Y_{1,n}|Y_{1,n-1}) P(Z_1|Y_{1,n}) P(X) P(Y_{2,1}|X) \\ &\quad \cdot P(Y_{2,2}|Y_{2,1}) ... P(Y_{2,n}|Y_{2,n-1}) P(Z_2|Y_{2,n})\} \\ &= \sum_X P(X) \sum_{Y_{1,1}} P(Y_{1,1}|X) \sum_{Y_{1,2}} P(Y_{1,2}|Y_{1,1}) \\ &\quad ... \sum_{Y_{1,n}} P(Y_{1,n}|Y_{1,n-1}) P(Z_1|Y_{1,n}) \sum_{Y_{2,1}} P(Y_{2,1}|X) \\ &\quad \cdot \sum_{Y_{2,2}} P(Y_{2,2}|Y_{2,1}) ... \sum_{Y_{2,n}} P(Y_{2,n}|Y_{2,n-1}) P(Z_2|Y_{2,n}) \\ &= \sum_X P(X) f(X, Z_1) g(X, Z_2),\end{aligned} \tag{26}$$

where $f(X = i, Z_1 = j)$ is the same as the ith row and jth col of matrix $\prod_{i=1}^{n+1} U_{1,i}$, and $g(X = i, Z_2 = j)$ is same as the ith row and jth col of matrix $\prod_{i=1}^{n+1} U_{2,i}$. When n goes to infinite, we can get that all elements in a special col of $\prod_{i=1}^{n+1} U_{1,i}$ or $\prod_{i=1}^{n+1} U_{2,i}$ tend to a same value by using lemma 2. It means that $f(X, Z_1)$ and $g(X, Z_2)$ are independent of X respectively. In other words $f(X, Z_1) \approx f_1(Z_1)$ and $g(X, Z_2) \approx g_1(Z_2)$. As a

result, when n goes to infinite, we can get:

$$\begin{aligned} P(Z_1, Z_2) &= \sum_X P(X) f(X, Z_1) g(X, Z_2) = \sum_X P(X) f_1(Z_1) g_1(Z_2) \\ &= f_1(Z_1) g_1(Z_2) \\ &= \left(\sum_X P(X) f_1(Z_1)\right)\left(\sum_X P(X) g_1(Z_2)\right) \\ &= \left(\sum_X P(X) f(X, Z_1)\right)\left(\sum_X P(X) g(X, Z_2)\right) \\ &= P(Z_1) P(Z_2). \end{aligned} \tag{27}$$

This means $Z_1 \perp Z_2$ as n tends to infinite.

ACKNOWLEDGMENT

The authors would also like to thank the reviewers for valuable comments and helpful guidance. This work is supported in part by the Natural Science Foundation of China (Nos. 61425025 and 61390515), and in part by the Human Brain Project of the European Union #604102 and #720270.

REFERENCES

[1] A. Pouget, J. Drugowitsch, and A. Kepecs, "Confidence and certainty: distinct probabilistic quantities for different goals," *Nature Neuroscience*, vol. 19, no. 3, pp. 366–374, 2016.

[2] 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.

[3] K. P. Körding and D. M. Wolpert, "Bayesian integration in sensorimotor learning," *Nature*, vol. 427, no. 6971, pp. 244–247, 2004.

[4] J. Austerweil, S. Gershman, J. Tenenbaum, and T. Griffiths, *Structure and flexibility in Bayesian models of cognition. In Oxford Handbook of Computational and Mathematical Psychology*. Oxford: Oxford University Press, 2015.

[5] R. P. Rao, "Bayesian computation in recurrent neural circuits," *Neural computation*, vol. 16, no. 1, pp. 1–38, 2004.

[6] ——, "Hierarchical Bayesian inference in networks of spiking neurons," in *Advances in neural information processing systems*, 2004, pp. 1113–1120.

[7] J. M. Beck and A. Pouget, "Exact inferences in a neural implementation of a hidden Markov model," *Neural Computation*, vol. 19, pp. 1344–1361, 2007.

[8] T. Ott and R. Stoop, "The neurodynamics of belief propagation on binary markov random fields," in *Advances in neural information processing systems*, 2006, pp. 1057–1064.

[9] Z. Yu, F. Chen, and J. Dong, "Neural network implementation of inference on binary markov random fields with probability coding," *Applied Mathematics and Computation*, vol. 301, pp. 193–200, 2017.

[10] D. George and J. Hawkins, "Belief propagation and wiring length optimization as organizing principles for cortical microcircuits," Technical report, Numenta, http://www. numenta. com, Tech. Rep., 2006.

[11] A. Steimer, W. Maass, and R. Douglas, "Belief propagation in networks of spiking neurons," *Neural Computation*, vol. 21, no. 9, pp. 2502–2523, 2009.

[12] S. Litvak and S. Ullman, "Cortical circuitry implementing graphical models," *Neural Computation*, vol. 21, no. 11, pp. 3010–3056, 2009.

[13] A. Pouget, J. M. Beck, W. J. Ma, and P. E. Latham, "Probabilistic brains: knowns and unknowns," *Nature Neuroscience*, vol. 16, no. 9, pp. 1170–1178, 2013.

[14] D. Koller and N. Friedman, *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

[15] C. Andrieu, N. De Freitas, A. Doucet, and M. I. Jordan, "An introduction to mcmc for machine learning," *Machine learning*, vol. 50, no. 1-2, pp. 5–43, 2003.

[16] J. Cheng and M. J. Druzdzel, "Ais-bn: An adaptive importance sampling algorithm for evidential reasoning in large Bayesian networks," *Journal of Artificial Intelligence Research*, vol. 13, pp. 155–188, 2000.

[17] L. Shi and T. L. Griffiths, "Neural implementation of hierarchical Bayesian inference by importance sampling," in *Advances in neural information processing systems*, 2009, pp. 1669–1677.

[18] D. L. Yamins and J. J. DiCarlo, "Using goal-driven deep learning models to understand sensory cortex," *Nature Neuroscience*, vol. 19, no. 3, pp. 356–365, 2016.

[19] Q. Liu, "Importance weighted consensus Monte Carlo for distributed Bayesian inference." in *UAI*, 2016.

[20] P. M. Krafft, J. Zheng, W. Pan, N. Della Penna, Y. Altshuler, E. Shmueli, J. B. Tenenbaum, and A. Pentland, "Human collective intelligence as distributed Bayesian inference," *arXiv preprint arXiv:1608.01987*, 2016.

[21] C. S. Thakur, S. Afshar, R. M. Wang, T. J. Hamilton, J. Tapson, and A. Van Schaik, "Bayesian estimation and inference using stochastic electronics," *Frontiers in Neuroscience*, vol. 10, 2016.

[22] L. Hasenclever, S. Webb, T. Lienart, S. Vollmer, B. Lakshminarayanan, C. Blundell, and Y. W. Teh, "Distributed Bayesian learning with stochastic natural gradient expectation propagation and the posterior server," *Journal of Machine Learning Research*, vol. 18, no. 106, pp. 1–37, 2017.

[23] T. Gollisch and M. Meister, "Eye smarter than scientists believed: neural computations in circuits of the retina," *Neuron*, vol. 65, no. 2, pp. 150–164, 2010.

[24] H. Liu, A. Rios-Navarro, D. P. Moeys, T. Delbruck, and A. Linares-Barranco, "Neuromorphic approach sensitivity cell modeling and FPGA implementation," in *International Conference on Artificial Neural Networks*. Springer, 2017, pp. 179–187.

[25] J. A. Lorteije, A. Zylberberg, B. G. Ouellette, C. I. De Zeeuw, M. Sigman, and P. R. Roelfsema, "The formation of hierarchical decisions in the visual cortex," *Neuron*, vol. 87, no. 6, pp. 1344–1356, 2015.