Physically Equivalent Magneto-Electric Nanoarchitecture for Probabilistic Reasoning

Santosh Khasanvis[1], Mingyu Li[1], Mostafizur Rahman[1], Mohammad Salehi-Fashami[2], Ayan K. Biswas[2], Jayasimha Atulasimha[2], Supriyo Bandyopadhyay[2], and Csaba Andras Moritz[1*]
Department of ECE, University of Massachusetts Amherst[1], Amherst, MA, USA
Virginia Commonwealth University[2], Richmond, VA, USA
Email: andras@ecs.umass.edu*

***Abstract*—Probabilistic machine intelligence paradigms such as Bayesian Networks (BNs) are widely used in critical real-world applications. However they cannot be employed efficiently for large problems on conventional computing systems due to inefficiencies resulting from layers of abstraction and separation of logic and memory. We present an unconventional nanoscale magneto-electric machine paradigm, architected with the principle of *physical equivalence* to efficiently implement causal inference in BNs. It leverages emerging straintronic magneto-tunneling junctions in a novel mixed-signal circuit framework for direct computations on probabilities, while blurring the boundary between memory and computation. Initial evaluations, based on extensive bottom-up simulations, indicate up to four orders of magnitude inference runtime speedup vs. best-case performance of 100-core microprocessors, for BNs with a million random variables. These could be the target applications for emerging magneto-electric devices to enable capabilities for leapfrogging beyond present day computing.**

Keywords—Bayesian networks; non-Boolean computing; mixed-signal; magnetic tunneling junctions; memory-in-computing

I. INTRODUCTION

Bayesian Networks (BNs) belong to a class of widely successful probabilistic formalism capable of reasoning under uncertainty, and are used for several real-world applications, e.g., gene association networks, text-classification, network threat monitoring, etc. A BN encodes knowledge of a domain in its structure (directed acyclic graph showing dependencies between variables) and parameters (conditional probability tables, CPTs, quantifying strength of relationships among variables). It can be used for expressing the strength of belief in the state of a system given some observations on its environment. This process of BN inference requires computation of belief (**BEL**), i.e. the probability of a hypothesis given evidence, and is performed via message propagation (likelihoods $\boldsymbol{\lambda}$ and priors $\boldsymbol{\pi}$ [1]) through the network using Pearl's Belief Propagation algorithm [1]. The key operations in a BN inference require (i) distributed storage of probabilities, and (ii) frequent arithmetic operations such as multiplication and addition on probabilities.

Conventional von Neumann architectures are not well suited because they (i) require emulation on Boolean logic framework, (ii) incorporate a limited number of arithmetic units (due to high implementation complexity) leading to serialized execution, (iii) do not support distributed local storage and processing, and (iv) use a radix-based representation of data which has no inherent fault-resilience.

This material is based upon work supported by the National Science Foundation grant no. 1407906 at UMass Amherst, and National Science Foundation grants ECCS-1124714, CCF-1216614 and CCF-1253370 at VCU.

Our objective is to architect an efficient machine for the causal reasoning framework using emerging nanotechnology. Rather than mimicking conventional computing mindset, our goal is to identify representations resulting in *physical equivalency* with the conceptual probabilistic framework across all layers from data representation to circuits and architecture. We leverage straintronic magneto-tunneling junction devices (S-MTJs) [2] as the physical layer, which are attractive due to their extremely low energy of switching and support for non-volatility.

II. PHYSICALLY EQUIVALENT ARCHITECTURE FOR BAYESIAN INFERENCE ENGINES

Since BNs operate on probabilities, we represent information as a non-Boolean flat probability vector [3], using n spatially distributed digits (p_1, p_2,…, p_n). Each digit p_i can take any one of k values, where k is the number states supported by the underlying physical device (e.g., for k=4 and $p_i \in \{0,1,2,3\}$). In this work, we focus on binary devices. The encoded probability $\mathbf{P} = \sum_{i=1}^{n} p_i/[n(k-1)]$. This representation also yields graceful degradation in case of faults. It is tied to the physical layer through S-MTJs (Fig. 1A-B), where the resistance state of each S-MTJ stores a probability digit.

At the circuit level, we use a physically equivalent mixed-signal framework for implementing Bayesian inference functions. It operates directly on probabilities, by converting them from discrete spatial vectors (resistance states of several S-MTJs) to equivalent analog current/voltage values, and circuit topology defines underlying analog computation. This circuit style is referred to as Probability Arithmetic Composers [3]. Bayesian inference operations are composed hierarchically using Elementary Probability Arithmetic Composers performing analog arithmetic (multiplication and addition), in contrast to emulation with Boolean logic gates (Fig. 1C). This leads to compact circuits, which are also capable of storing state information due to S-MTJ non-volatility. Analog outputs from computations are converted back to probability vectors using Decomposer circuits [3].

Building on this circuit style we implement a Physically Equivalent Architecture for Reasoning under Uncertainty (PEAR) that intrinsically supports BNs (Fig. 1D-E). A departure from von Neumann mindset, it uses a distributed Bayesian Cell (BC) framework where a single BC or a cluster of BCs can be used to directly map a BN node in hardware for inference. BN parameters (CPTs) and state information (likelihood, belief and prior vectors) are stored in the

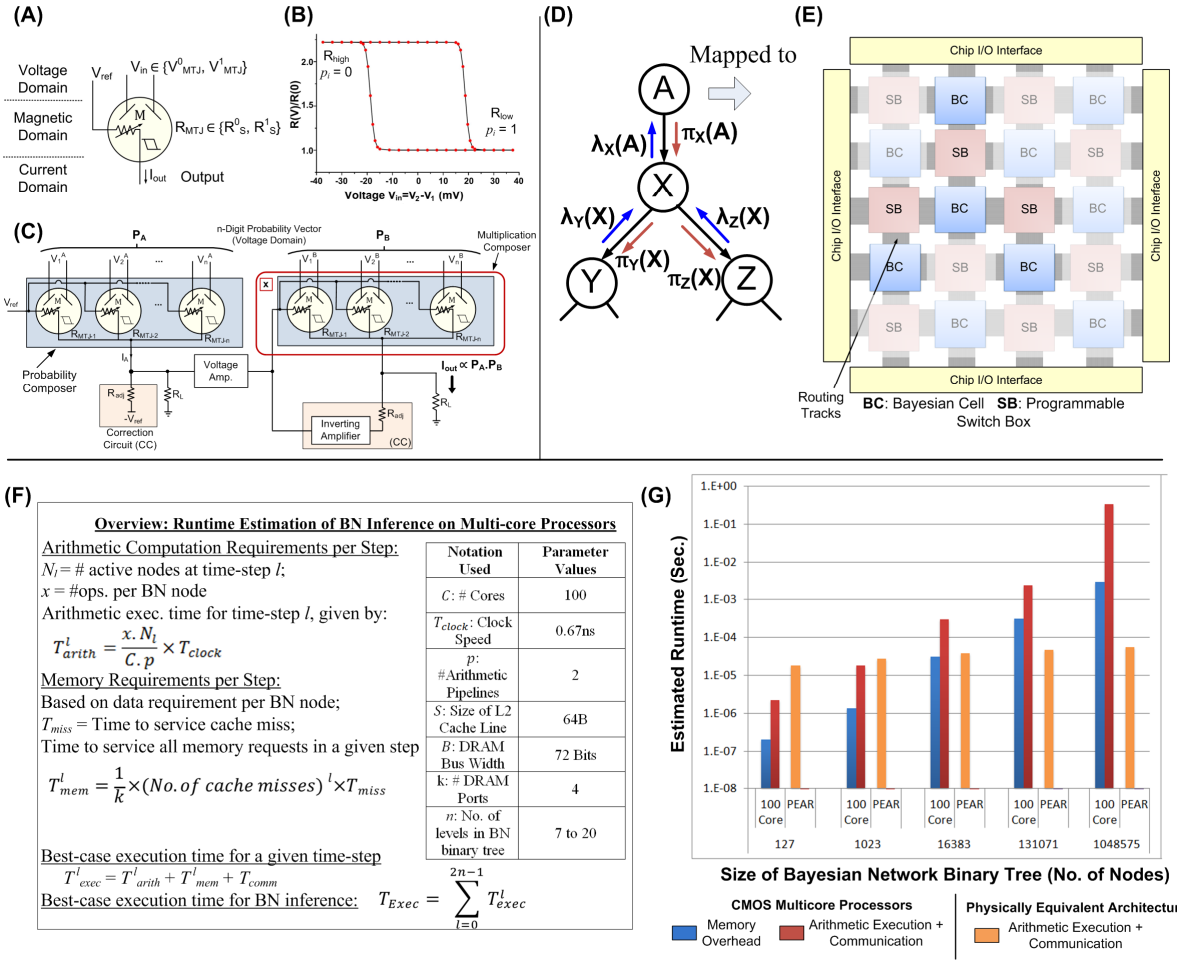

Fig. 1. (A) Non-volatile SMTJ schematic; (B) S-MTJ device characteristics showing hysteresis and probability digit encoding in resistance state; (C) Example Probability Arithmetic Composer using S-MTJs for multiplication operation [3]. Composers for other operations are discussed in detail in ref. [3]; (D) Part of a BN showing message propagation; (E) Programmable physically equivalent architecture, PEAR, for mapping BNs in hardware: BCs map nodes directly in hardware - Example mapping shown for the BN graph from (D), where grayed cells are inactive; (F) Analytical model overview for estimating inference runtime on CMOS 100-core processor. Specifications are from ref. [4]; (G) Runtime Comparison.

Composer circuits themselves in BCs, obviating the need for external memory. BCs are interconnected through CMOS metal routing stack for message propagation, made programmable through reconfigurable switch-boxes (using transistors gated by S-MTJs) for mapping arbitrary BNs. An Activity Controller can be used for power mitigation by switching off the cell when idle.

III. Evaluation and Conclusion

Extensive LLG simulations for device characteristics (Fig. 1B) and HSPICE circuit simulations were performed to evaluate PEAR. We use a binary tree with four states for each node as an example BN, and scale the number of variables in the order of a 100 to a million. We use analytical model (Fig. 1F) for runtime estimation on CMOS 100-core processors [4], which represent the best-in-breed for von Neumann architectures. Our multi-core processor analysis is under ideal assumptions for performance in general, and is reflective of best-case scenario. Our evaluations (Fig. 1G) indicate that PEAR can provide up to 4 orders of magnitude performance speedup over 100-core processors, in supporting BNs with large problem sizes involving a million variables. It shows promise for realizing highly efficient reasoning machines at nanoscale.

References

[1] J. Pearl, *Probabilistic reasoning in intelligent systems: Networks of plausible inference*, San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1988.

[2] A. K. Biswas, S. Bandyopadhyay and J. Atulasimha, "Energy-efficient magnetoelastic non-volatile memory," Appl. Phys. Lett., 104, 232403, 2014.

[3] S. Khasanvis, et al. , "Self-similar magneto-electric nanocircuit technology for probabilistic inference engines," *IEEE Trans. on Nanotechnology*, Special Issue on Cognitive Computing with Nanotechnology, accepted. Preprint: http://arxiv.org/abs/1504.04056

[4] C. Ramey, "TILE-Gx100 manycore processor: Acceleration interfaces and architecture", Aug. 2011, Tilera Corporation. Available Online: http://www.hotchips.org/wp-content/uploads/hc_archives/hc23/HC23.18.2-security/HC23.18.220-TILE-GX100-Ramey-Tilera-e.pdf