

---

# Energy-efficient Machine Learning in Silicon: A Communications-inspired Approach

---

Naresh Shanbhag

SHANBHAG@ILLINOIS.EDU

University of Illinois at Urbana-Champaign, Urbana, IL 61801 USA

## Abstract

This position paper advocates a communications-inspired approach to the design of machine learning systems on energy-constrained embedded ‘always-on’ platforms. The communications-inspired approach has two versions - 1) a deterministic version where existing low-power communication IC design methods are repurposed, and 2) a stochastic version referred to as Shannon-inspired *statistical information processing* employing information-based metrics, *statistical error compensation* (SEC), and retraining-based methods to implement ML systems on stochastic circuit/device fabrics operating at the limits of energy-efficiency. The communications-inspired approach has the potential to fully leverage the opportunities afforded by ML algorithms and applications in order to address the challenges inherent in their deployment on energy-constrained platforms.

## 1. Introduction

Machine learning (ML)-based systems are transforming the way we live and interact with the world around us. In many tasks, such as those in computer vision, machines have begun to exceed human performance (Silver et al., 2016). However, machines have much catching up to do when energy costs are accounted for. While it is difficult to accurately estimate the energy cost of the AlphaGo system developed by Google DeepMind when it beat the human champion recently in the ancient game of Go, one can safely assume that the machine consumed about four-orders-of-magnitude higher power (1202 CPUs and 176 GPUs (Silver et al., 2016)) as compared to the nominally quoted power of 20 W for the human brain. If ML systems need to become pervasive in our lives then it is imper-

ative that this energy cost be significantly reduced. The availability of such low-energy realizations of ML systems will enable its deployment on embedded platforms such as biomedical devices, wearables, autonomous vehicles, IoT and many others. Not surprisingly, a number of integrated circuit (IC) implementations of ML kernels and algorithms have appeared recently (Chen et al., 2016; Kaul et al., 2016; Park et al., 2016) that have set energy-efficiency records. However, much work still remains to be done as the energy gap between these realizations and that achieved by the human brain remains huge. In particular, the search for minimum energy realizations of ML systems needs to be done systematically. The low-energy ML design space is complex as it encompasses deeply intertwined issues at the algorithmic, architectural, circuit and the device level. The mainstream approach today is

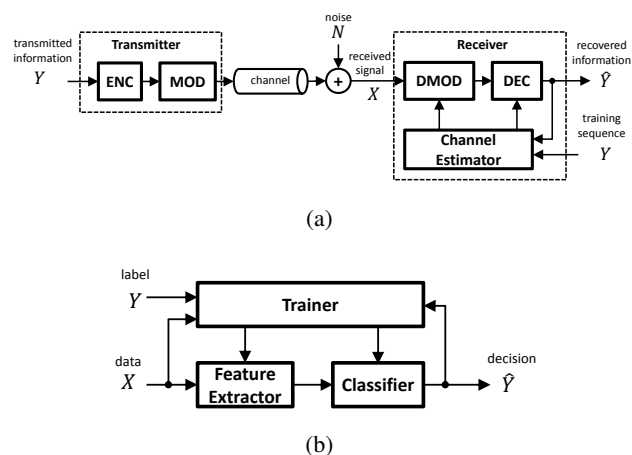


Figure 1. Viewing a communication receiver as an inference system: (a) the communication link, and (b) a ML system.

to treat the problem of energy-efficient ML implementation as yet another problem in energy-efficient computing. We believe that there are substantial gains to be made if

---

This work was supported in part by Systems on Nanoscale Information fabriCs (SONIC), one of the six SRC STARnet Centers, sponsored by MARCO and DARPA.

one were to repurpose the vast body of knowledge accumulated over two and a half decades by the designers of low-power communication and signal processing systems and ICs (A.P. Chandrakasan & Brodersen, 1992; Shanbhag, 1998; Parhi, 1999). This position paper makes the case for employing a communications-inspired approach in order to explore the design of energy-efficient ML in nanoscale silicon CMOS and emerging beyond CMOS device fabrics.

The communications-inspired approach is based on drawing parallels between a communication receiver and an inference kernel as shown in Fig. 1. A communication receiver infers the transmitted symbols  $Y$  from the received signal  $X$ , much as a ML system infers the class label  $Y$  from the observed data  $X$ . In both systems, the process of inference needs to be accomplished in the presence of random noise and incomplete data. Both systems need an element of learning/training to be present in order to incorporate time-varying/unknown data statistics/model into the decision making process. Communication receivers commonly employ statistical estimation procedures to learn the channel parameters, which are then employed for data recovery. Furthermore, the stochastic gradient descent (SGD) (Mathews & Xie, 1993; Keuper & Pfreundt, 2015) is commonly employed in both systems due to its ease of implementation and robustness. There is one key difference between the two systems though. In communication systems, the data  $X$ 's statistics can be *engineered* via proper coding and modulation in the transmitter. This allows such receiver to operate with well-structured signal, channel and noise models, which lowers its complexity and energy consumption, while enhancing its accuracy. This flexibility may not be present in general ML scenarios. Nevertheless, the similarities between the two are substantial enough to warrant a closer look at low-power communication receiver design techniques and see which ones might be repurposed for ML systems.

In the discussion above, one assumes a deterministic circuit fabric. Recent IC implementations (Chen et al., 2016; Kaul et al., 2016; Park et al., 2016) do in fact fit this model. However, this assumption can be relaxed in case of ML systems due to their inherent ability to operate in the presence of incomplete or noisy data. This ability can be leveraged to address the statistical behavior of circuit/device fabrics that arises when these operate at the limits of energy efficiency. Such ultimate low-energy fabrics is referred to as *stochastic fabrics* or *low-SNR circuit fabrics*. Indeed, statistical behavior in such fabrics can arise when:

- operating at very low voltages (Dreslinski et al., 2010) or low area (Roy et al., 2013), both of which result in computational errors, and/or
- designing systems with emerging devices (Roy et al., 2013; Wei et al., 2013) which tend to be intrinsically

statistical in nature due to nanoscale imperfections such as variations and defects, and/or

- embedding computation into memory (in-memory computing (Kang et al., 2014)) and sensing (in-sensor computing (Hu et al., 2012)) substrates in order to drastically reduce/eliminate data movement.

We refer to such ultimate low-energy fabrics as *stochastic fabrics*. The statistical behavior of stochastic fabrics needs to be compensated for much as a communication receiver compensates for the statistical behavior of the channel. The communications-inspired view opens up the possibility of taking the connections between ML and communications to another level by treating the circuit fabric itself as a noisy channel on which to extract information from data. We refer to this second approach as Shannon-inspired *statistical information processing* (Shanbhag et al., 2010). Statistical information processing involves the use of information-based metrics, *statistical error compensation* (SEC) (Hegde & Shanbhag, 2001), and retraining approaches such as data-driven hardware resiliency (DDHR) (Wang et al., 2015) to enhance robustness. One intellectually satisfying aspect of statistical information processing is the potential for developing a comprehensive foundation for reliable information processing on stochastic fabrics much as Claude Shannon (Shannon, 1948) established one for reliable communications over a noisy channel. Such a foundation needs to provide fundamental bounds on the information processing capacity, energy-efficiency, robustness, as well as practical design techniques, e.g., SEC and DDHR, to approach these bounds.

This paper advocates a communications-inspired approach to the design of energy-efficient ML systems on both deterministic and stochastic fabrics. Doing so will bring together methodologies such as low-power signal processing algorithms and architectures (Parhi, 1999), algorithm transforms (Shanbhag, 1998), low-power integrated circuit (IC) design (A.P. Chandrakasan & Brodersen, 1992), information-based design metrics, *statistical error compensation* (SEC) and others to systematically explore the design space in order to determine minimum energy realizations.

## 2. Machine Learning on Deterministic Fabrics

The design of communication receiver ICs begins with algorithm design employing statistical signal processing techniques such as estimation and detection to meet a specific system design metric such as the bit-error rate (BER)  $p_e = P\{Y \neq \hat{Y}\}$  (see Fig. 1). The use of an information-based metric (BER) and its intrinsically statistical nature makes it possible to reduce algorithmic com-

plexity right from the start. Redundant algorithmic operations are eliminated or substituted with approximate ones so as to leave the BER unaltered. Machine learning systems employ an accuracy metric  $p_{det}$  the probability of detection, and therefore can benefit from such approximations. Indeed, “approximate computing” (Venkataramani et al., 2015) strives to build a methodology to systematize and repurpose these concepts which are well-known and well-practiced for decades by communication IC designers. The result of this step is a floating-point algorithm meeting the system requirements on BER and other metrics.

Next, *fixed-point analysis* is employed to minimize the precision of computation and storage. Indeed, minimizing precision (Gupta et al., 2015) is an effective approach to reduce energy. The goal of this step is to minimize the BER difference between the floating-point and a fixed-point algorithm. Precisions is typically obtained via trial-and-error. Insights on what algorithmic aspects determine the precision tend to be lost in this process. However, for communications and ML algorithms, it is possible to obtain analytical bounds on precision. For example, the bounds on the precision  $B_{WUD}$  of the weight-update unit of the popular least mean-squared (LMS) algorithm (Goel & Shanbhag, 1998) is given by:

$$B_{WUD} \geq \frac{1}{2} \log_2 \left( \frac{1}{\mu^2 \sigma_x^2 \sigma_y^2} \right) + \frac{SNR_{fl}(dB)}{6} \quad (1a)$$

where  $\mu$  is the step-size,  $\sigma_x^2$  and  $\sigma_y^2$  are variances of the input  $X$  and desired signal  $Y$ , respectively, and  $SNR_{fl}(dB)$  is the SNR of the floating point algorithm in dBs. Minimum precision requirements are thus obtained without resorting to expensive simulations. In a similar fashion, it is possible to obtain bounds for other SGD-based on-line learning algorithms.

The fixed-point algorithm can be described using a data flow-graph (DFG) or a control and data flow graph (CDFG). An almost infinite variety of architectures can be systematically obtained from a DFG using algorithm transforms (Parhi, 1999) such as unfolding, folding, pipelining, systolization, among others. ML algorithms tend to have a regular DFG (see Fig. 2). This opens up the possibility of realizing *systolic architectures* (Kung, 1982) for many ML algorithms. Some work already exists (Jones et al., 1994; Kung & Hwang, 1989). Systolic architectures are regular, have local interconnections, and can be designed to minimize data movement. The process of mapping a regular DFG to a systolic architectures involves the selection of a *processor vector*  $\mathbf{p}$ , the *iteration vector*  $\mathbf{d}$  and the *schedule vector*  $\mathbf{s}$ , satisfying the constraints  $\mathbf{p}^T \mathbf{d} = 0$ ,  $\mathbf{s}^T \mathbf{d} \neq 0$ , and implying that the DFG node  $\mathbf{v}$  is mapped to processor  $\mathbf{p}^T \mathbf{v}$  in the cycle  $\mathbf{s}^T \mathbf{v}$ . Indeed, one can derive the recently proposed architectures (Chen et al., 2016; Murmann et al., 2015) by formulating the DFG of a convolutional neural

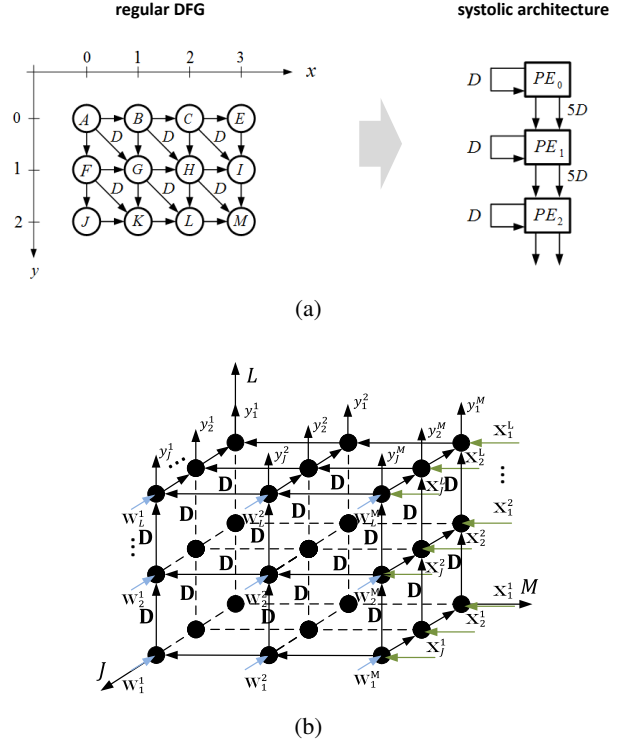


Figure 2. Systolization: (a) a regular DFG mapped to a systolic architecture via specific choices of vectors  $\mathbf{p} = [1 \ 0]^T$ ,  $\mathbf{d} = [0 \ 1]^T$  and  $\mathbf{s} = [1 \ 0]^T$ , where  $D$  is a 1-sample delay element, and (b) the DFG of the C-layer of a CNN with each node being a dot-product computation.

network (CNN) (LeCun et al., 1998) (see Fig. 2(b)), and assigning appropriate values to  $\mathbf{p}$ ,  $\mathbf{d}$ , and  $\mathbf{s}$ , along with the folding transform. These design methodologies for communication ICs can be repurposed for the design of energy-efficient ML systems in silicon.

### 3. Machine Learning on Stochastic Fabrics

The communications-inspired approach presents a unique opportunity when implementing ML on deeply scaled nanofabrics that operate at the limits of energy efficiency where a transition into non-determinism occurs. For example, near/subthreshold voltage (Dreslinski et al., 2010) operation in CMOS results approximately  $10\times$  reduction in energy but at the expense of up to  $20\times$  increase in delay variations. This variability eventually translates into observable errors in computation, storage, and communications. We refer to such circuit and device substrate as *stochastic fabrics*, and the errors themselves as *fabric noise*. ML algorithms’ intrinsic robustness to data noise enables it to absorb the impact of fabric noise. This feature, referred to popularly as ‘error-tolerance’, can be exploited to some extent by approaches such as approximate

computing (Venkataramani et al., 2015) as well. However, it is possible to reduce the energy consumption even further by operating the circuit fabric at a point where the intrinsic error-tolerance of the algorithm is exceeded. At this point, corrective measures, i.e., error compensation methods, need to be incorporated. Conventional fault-tolerance techniques such as  $N$ -modular redundancy are ineffective as these have a high energy-cost, and do not account for the unique attributes of ML algorithms. A Shannon-inspired approach to error compensation turns out to be most effective.

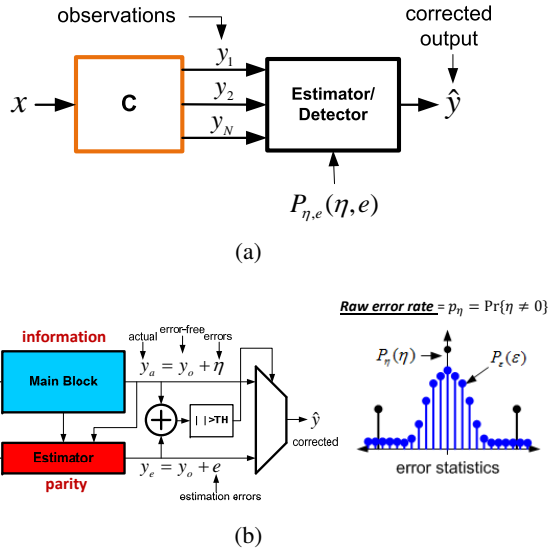


Figure 3. Shannon-inspired statistical error compensation (SEC): (a) a general framework, and (b) algorithmic noise-tolerance (ANT).

In the past, we have proposed the notion of treating the stochastic circuit fabric as a noisy communication channel (Shanbhag, 1996) and develop Shannon-inspired statistical error compensation (SEC) techniques (see Fig. 3(a))(Hegde & Shanbhag, 2001; Shim et al., 2004; Varatkar et al., 2010) to compensate for the resulting errors at the algorithmic and architectural levels. Prototype ICs (see Fig. 4) demonstrating these ideas have been implemented. These demonstrate that computational error rates, defined as the probability of an incorrect output, of 60% (Abdallah & Shanbhag, 2013) and in specific cases (see Fig. 4(b)), up to 80% (Kim et al., 2015) can be compensated for by applying techniques based on statistical estimation and detection. SEC techniques have shown to result in energy savings ranging from  $3\times$ -to- $6\times$  over designs that work on deterministic fabrics.

The ability to compensate for such high computational error rates motivates the idea of *in-situ data analytics*, where computation is deeply embedded into the same substrate where data is stored or being acquired, e.g., *in-memory*

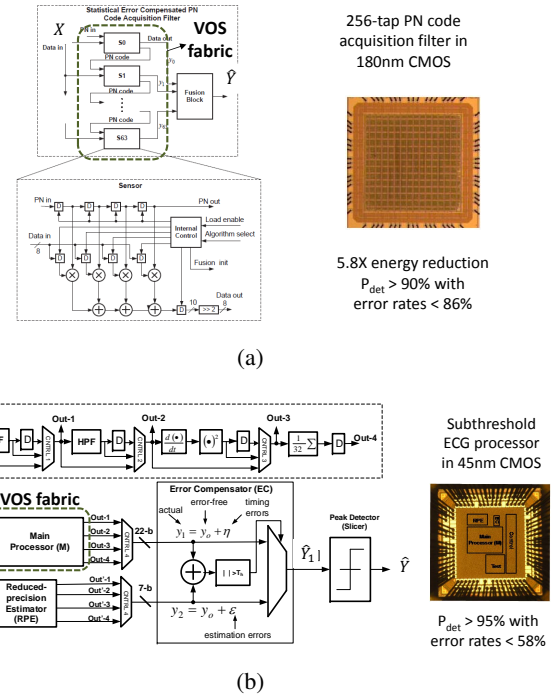


Figure 4. Statistical error compensation (SEC) based IC prototypes: (a) 256-tap PN code acquisition in 180 nm CMOS, and (b) ECG processor in 45 nm CMOS.

(Kang et al., 2014) and *in-sensor* computing (Hu et al., 2012). Such substrates are not particularly well-suited for deterministic von Neumann style computing but fits the Shannon-inspired style. Thus, SEC leverages Shannon theory to develop techniques to compensate for errors that cannot be absorbed by the intrinsic error-tolerance of the algorithm. This key aspect distinguishes it from techniques that seek to work within the error-tolerance envelope of the algorithm. SEC techniques can be made adaptive in order to track variations in the data and error statistics. ML-based SEC techniques can also be developed.

Another approach is DDHR (Wang et al., 2015) that employs retraining to obtain parameters of the algorithm to compensate for both data and fabric noise. Both SEC and DDHR leverage the statistical nature of system and application metrics, and may even be combined in a synergistic fashion.

## 4. Summary

ML systems have unique properties that it shares with communication systems. There is much to be gained by exploiting the connections between the two when exploring energy efficient on-device implementations of ML systems.

## References

- Abdallah, R. A. and Shanbhag, N. R. An Energy-Efficient ECG Processor in 45-nm CMOS Using Statistical Error Compensation. *IEEE Journal of Solid-State Circuits*, 48 (11):2882–2893, Nov 2013. ISSN 0018-9200.
- A.P. Chandrakasan, S. Sheng and Brodersen, R.W. Low-Power CMOS digital design. *IEEE Journal of Solid-State Circuits*, 27(4):473 – 484, April 1992.
- Chen, Y. H., Krishna, T., Emer, J., and Sze, V. Eyeriss: An energy-efficient reconfigurable accelerator for deep convolutional neural networks. In *ISSCC 2016*, pp. 262–263, Jan 2016.
- Dreslinski, R. G., Wieckowski, M., Blaauw, D., Sylvester, D., and Mudge, T. Near-Threshold Computing: Reclaiming Moore’s Law Through Energy Efficient Integrated Circuits. *Proceedings of the IEEE*, 98(2):253–266, Feb 2010. ISSN 0018-9219.
- Goel, M. and Shanbhag, N. R. Finite-precision analysis of the pipelined strength-reduced adaptive filter. *IEEE Transactions on Signal Processing*, 46(6):1763–1769, Jun 1998.
- Gupta, S., Agrawal, A., Gopalakrishnan, K., and Narayanan, P. Deep Learning with Limited Numerical Precision. In *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, pp. 1737–1746, 2015.
- Hegde, R. and Shanbhag, N. R. Soft digital signal processing. *IEEE Transactions on VLSI Systems*, pp. 813–823, December 2001.
- Hu, Y., Rieutort-Louis, W., Sanz-Robinson, J., Song, K., Sturm, J. C., Wagner, S., and Verma, N. High-resolution sensing sheet for structural-health monitoring via scalable interfacing of flexible electronics with high-performance ICs. In *2012 Symposium on VLSI Circuits (VLSIC)*, pp. 120–121, June 2012.
- Jones, S. R., Sammut, K. M., and Hunter, J. Learning in linear systolic neural network engines: analysis and implementation. *IEEE Transactions on Neural Networks*, 5 (4):584–593, Jul 1994.
- Kang, M., Keel, M. S., Shanbhag, N. R., Eilert, S., and Curewitz, K. An energy-efficient VLSI architecture for pattern recognition via deep embedding of computation in SRAM. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 8326–8330, May 2014.
- Kaul, H., Anders, M. A., Mathew, S. K., Chen, G., Satpathy, S. K., Hsu, S. K., Agarwal, A., and Krishnamurthy, R. K. A 21.5M-query-vectors/s 3.37nJ/vector reconfigurable k-nearest-neighbor accelerator with adaptive precision in 14nm tri-gate CMOS. In *2016 IEEE International Solid-State Circuits Conference (ISSCC)*, pp. 260–261, Jan 2016.
- Keuper, J. and Pfreundt, F. Asynchronous Parallel Stochastic Gradient Descent: A Numeric Core for Scalable Distributed Machine Learning Algorithms. In *Proceedings of the Workshop on Machine Learning in High-Performance Computing Environments, MLHPC ’15*, pp. 1:1–1:11, New York, NY, USA, 2015. ISBN 978-1-4503-4006-9.
- Kim, E. P., Baker, D. J., Narayanan, S., Shanbhag, N. R., and Jones, D. L. A 3.6-mW 50-MHz PN Code Acquisition Filter via Statistical Error Compensation in 180-nm CMOS. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 23(3):598–602, March 2015. ISSN 1063-8210.
- Kung, H. T. Why systolic architectures? *Computer*, 15(1): 37–46, Jan 1982. ISSN 0018-9162.
- Kung, S. and Hwang, J. A Unified Systolic Architecture for Artificial Neural Networks. *Journal of Parallel and Distributed Computings*, 6:358–387, April 1989.
- LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 1998.
- Mathews, V. J. and Xie, Z. A stochastic gradient adaptive filter with gradient adaptive step size. *IEEE Transactions on Signal Processing*, 41(6):2075–2087, Jun 1993. ISSN 1053-587X.
- Murmann, B., Bankman, D., Chai, E., Miyashita, D., and Yang, L. Mixed-signal circuits for embedded machine-learning applications. In *2015 49th Asilomar Conference on Signals, Systems and Computers*, pp. 1341–1345, Nov 2015.
- Parhi, K. K. *VLSI Digital Signal Processing Systems: Design and Implementation*. Wiley, 1999.
- Park, S., Choi, S., Lee, J., Kim, M., Park, J., and Yoo, H. J. A 126.1mW real-time natural UI/UX processor with embedded deep-learning core for low-power smart glasses. In *2016 IEEE International Solid-State Circuits Conference (ISSCC)*, pp. 254–255, Jan 2016.
- Roy, K., Sharad, M., Fan, Deliang, and Yogendra, K. Beyond charge-based computation: Boolean and non-Boolean computing with spin torque devices. In *Low Power Electronics and Design (ISLPED), 2013 IEEE International Symposium on*, pp. 139–142, Sept 2013.

- Shanbhag, N. R. Lower bounds on power-dissipation for DSP algorithms. In *Low Power Electronics and Design, 1996., International Symposium on*, pp. 43–48, Aug 1996. doi: 10.1109/LPE.1996.542728.
- Shanbhag, N. R. Algorithm transformation techniques for low-power wireless VLSI systems design. *International Journal of Wireless Information Networks*, 5:147 – 171, 1998.
- Shanbhag, N. R., Abdallah, R. A., Kumar, R., and Jones, D. L. Stochastic computation. In *Design Automation Conference (DAC), 2010 47th ACM/IEEE*, pp. 859–864, June 2010.
- Shannon, C. A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423, 1948.
- Shim, B., Sridhara, S., and Shanbhag, N. R. Reliable low-power digital signal processing via reduced precision redundancy. *IEEE Transactions on VLSI*, 12(5):497 – 510, May 2004.
- Silver, D., Huang, A., and et al. Mastering the game of go with deep neural networks and tree search. *Nature*, 529: 484–503, 2016.
- Varatkar, G. V., Narayanan, S., Shanbhag, N. R., and Jones, D. L. Stochastic networked computation. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 18(10):1421–1432, Oct 2010. ISSN 1063-8210.
- Venkataramani, S., Chakradhar, S. T., Roy, K., and Raghunathan, A. Approximate computing and the quest for computing efficiency. In *2015 52nd ACM/EDAC/IEEE Design Automation Conference (DAC)*, pp. 1–6, June 2015. doi: 10.1145/2744769.2744904.
- Wang, Z., Lee, K. H., and Verma, N. Overcoming Computational Errors in Sensing Platforms Through Embedded Machine-Learning Kernels. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 23(8):1459–1470, Aug 2015.
- Wei, H., Shulaker, M., and et al. Carbon nanotube circuits: Opportunities and challenges. In *Design, Automation Test in Europe Conference Exhibition (DATE), 2013*, pp. 619–624, March 2013.