On Analog Computing with Applications for Deep Learning

Isaiah R. Harville
University of Alabama in Huntsville
Huntsville, Alabama, USA
ih0021@uah.edu

***Abstract*—Analog computers were mostly replaced by digital architectures in the 1960s. However, with the rise of Machine Learning, specifically Deep Learning, analog computing is beginning to come back due to its unique methods of processing information. This paper does not propose new research but instead aims to survey existing research to deepen the conversation and substantiate arguments for analog computing as the future of Artificial Intelligence and Machine Learning.**

***Index Terms*—Computer Architecture, Machine Learning, Deep Learning, Analog, Digital, Computing**

I. Introduction

Recent advancements in digital architecture, specifically in Graphics Processing Units (GPUs), have led to substantial increases in computing performance, with some GPUs achieving impressive capabilities while consuming only 300 watts. However, these digital systems still fall short compared to their analog counterparts. The human brain, for instance, is an analog computer that operates at about 20 watts and can still perform complex tasks such as advanced mathematics but is extended into creative arts like music composition. Digital architectures are not naturally occurring and will never entirely replace analog architecture in applications that require real-time response.

This paper serves as an academic review, analyzing existing research on analog computing in the context of Deep Learning applications. Unlike digital computing, which processes discrete data, analog computing operates with continuous data and yields faster, energy-efficient computations. This characteristic makes it uniquely suited to handling the vast datasets and complex mathematical computations typical in Deep Learning. The goal of this paper is to emphasize the importance of reevaluating a technology previously considered obsolete objectively.

The following sections of this paper cover several key areas:

- Section II discusses various deep learning networks and architectures.
- Section III presents an overview of analog computer architecture, detailing its inherent qualities and limitations and a brief discussion on hybrid architecture.
- Section IV explores the benefits of analog computing by detailing areas where it can outperform digital architectures.
- Section V concludes the paper.

By expanding the discussion on existing studies and their implications, this paper aims to generate curiosity and excitement around analog computing and affirm its role in the future of Artificial Intelligence and Machine Learning.

II. Deep Learning Networks

Deep Learning (DL) has seen remarkable advancements at an unbelievable rate. In the pursuit of maximizing speed, accuracy, and the security of DL, the number of network architectures has grown exponentially. The focus of this paper is not directly on Deep Learning but more on how it can be changed with the applications of Analog Computing. This section will provide a brief overview of common architectures. There are three main categories of Deep Neural Networks (DNNs): Fully Connected Networks (FCN), Convolutional Neural Networks (CNN), and Recurrent Neural Networks (RNN). The information in the following subsections is primarily drawn from [1].

A. Fully Connected Networks

Fully Connected networks are the ”simplest” form of deep neural networks. In these networks, every neuron in one layer is connected to every neuron in the subsequent layer. The mathematical representation of a layer in an FCN can be given by the equation:

$$\mathbf{y} = \sigma(\mathbf{W}\mathbf{x} + \mathbf{b}) \tag{1}$$

where $\mathbf{x}$ is the input vector, $\mathbf{W}$ represents the weight matrix, $\mathbf{b}$ is the bias vector, and σ denotes the activation function. This architecture becomes quite computationally expensive as the complexity increases by adding more layers or feeding larger quantities of data.

B. Convolutional Neural Networks

Convolutional Neural Networks are particularly effective for processing data with significant depth[1]; they specialize in reducing dimensionality without losing information. CNNs utilize layers of kernels that convolve over the input matrix to capture local and global features. The operation of a convolution layer can be described by:

$$\mathbf{Z} = \mathbf{K} * \mathbf{X} + \mathbf{b} \tag{2}$$

[1]Such as images or audio for Natural Language Processing

Submitted as the final paper for Introduction to Digital Computer Architecture, CS-413, at The University of Alabama in Huntsville.

where $*$ denotes the convolution operation between the kernel $\mathbf{K}$ and the input $\mathbf{X}$, and $\mathbf{b}$ is the bias. CNNs typically include pooling layers that reduce dimensions and parameters by applying operations such as max or average pooling.

C. Recurrent Neural Networks

Recurrent Neural Networks are designed to process sequential data and have yielded excellent results when applied to modulation classification in the Radio Frequency Machine Learning domain. Unlike FCN or CNN, RNNs have connections that loop back on themselves, allowing information from previous inputs to influence later decisions. The basic RNN cell can be described by the equations:

$$\mathbf{h}_t = \sigma(\mathbf{W}_{hh}\mathbf{h}_{t-1} + \mathbf{W}_{hx}\mathbf{x}_t + \mathbf{b}_h) \tag{3}$$

$$\mathbf{y}_t = \phi(\mathbf{W}_{yh}\mathbf{h}_t + \mathbf{b}_y) \tag{4}$$

where $\mathbf{x}_t$ is the input at time t, $\mathbf{h}_t$ is the hidden state at time t, $\mathbf{y}_t$ is the output, σ and ϕ are activation functions, and $\mathbf{W}_{hh}$, $\mathbf{W}_{hx}$, $\mathbf{W}_{yh}$, $\mathbf{b}_h$, and $\mathbf{b}_y$ are parameters of the model.

III. Analog Computer Architecture

A. Qualities

1) Processing: Digital systems require a digitalization process on all incoming signals, which adds significantly to complexity and creates a potential for data loss. However, with analog computing, there is no conversion between data formats. Analog computers can perform complex mathematical operations far more naturally than digital systems and don't require quantization or discretization of data; everything can be performed in place. Eliminating the shift of data around in memory can increase throughput drastically.

2) Energy Efficiency and Speed: Analog devices typically consume less power than their digital counterparts since they operate directly on the signals without needing complex conversion circuits and high-frequency clocking mechanisms. Analog computers also excel in parallelization. Most importantly, analog computing allows for real-time data processing without the latency introduced by the digitization process, making it exceptionally fast for training neural networks.

B. Limitations of Analog Architectures

While analog computing offers significant benefits in terms of processing speed and energy efficiency, there are still some drawbacks. Analog systems lack the precision of digital systems[2], leading to errors and instability in computations, particularly problematic in applications requiring high accuracy. Furthermore, analog components are typically less flexible and programmable than their digital counterparts. Once an analog system is physically configured, modifying it can be time-intensive, posing significant limitations for iterative development processes.

One critical issue is the integration of analog systems with existing digital infrastructure. Most modern computing environments and standards are built around digital technology, which makes the integration of analog systems complex. Systems requiring a need for data interfacing could entirely eliminate the benefits found in analog processing if done incorrectly.

While these challenges shouldn't be diminished, the pursuit of overcoming these limitations is an active area of research. One excellent study, in particular, presents interesting methods and proposes a method of dynamic precision [3]. Continuing to address these issues will be crucial to enable analog computers to fulfill their potential to revolutionize Deep Learning.

C. Integration with Digital Computers

While analog systems offer speed and energy efficiency, digital systems provide precision and programmability. By integrating the two, we create hybrid systems that excel in performance, flexibility, and functionality. The concept of combining analog and digital computers to utilize accuracy and efficiency is not new [4]. One approach to integrating analog and digital systems is the development of hybrid architectures where analog devices perform rapid, energy-efficient computations and digital components ensure accuracy and manage control logic. For instance, analog components can be used for high-speed data processing and initial calculations, while digital systems can perform final adjustments, error corrections, and data formatting. This approach leverages the analog's natural ability to handle noisy, real-world data and the digital's capacity to perform precise, error-free calculations. Several studies have demonstrated promising results. For example, projects involving neuromorphic computing, where analog components mimic the behavior of neurons for processing neural network implementations, have shown how analog elements can accelerate computation while digital elements handle tasks like learning algorithm adjustments and memory management [5].

IV. Benefits of Analog Computing

Analog computing architectures align much closer to the Deep Learning problem scope than their digital counterparts. The in-place nature of analog signal processing eliminates many of the discretization and quantization errors inherent in digital systems, allowing deep learning models to train more effectively and efficiently. This direct processing capability, with lower energy consumption and faster operation times, makes analog a promising choice for the next generation of AI/ML hardware.

Analog Computing is accompanied by imprecision due to noise and other factors. However, previous research [2] from IBM Research and IBM Watson AI shows that deep learning algorithms are resilient to randomness or uncertainty and, therefore, allow for a trade-off between algorithmic accuracy and numerical precision. This suggests that not all deep learning tasks require the highest level of precision that digital computing, through Graphics Processing Units (GPUs), typically provides. Pursuing maximum accuracy can lead to increased computational demands and energy consumption

[2]This is less relevant in Deep Learning applications, discussed in [2]

without proportionate gains in performance. In these cases, the excessive focus on precision can be considered unnecessary and inefficient, allowing for the use of analog architectures.

A. Protonic Programmable Resistors for Ultrafast Analog Computing

Recent advancements in nanoscale protonic programmable resistors offer substantial performance improvements over traditional components. Developed by researchers from the Massachusetts Institute of Technology and the MIT-IBM Watson AI Lab, these silicon-compatible devices are approximately 1,000 times smaller than biological cells and can operate significantly faster than conventional neurons and synapses. Utilizing high electric fields, these devices facilitate rapid proton intercalation within solid-state structures, achieving operational speeds in the order of nanoseconds at room temperature. This marks a considerable enhancement over biological counterparts. This was achieved through an ingenious use of nanoscale phosphosilicate glass as a protonic solid electrolyte. Together with a WO3 active channel and a Pd hydrogen reservoir, these resistors enable fast, reversible[3], and energy-efficient modulation of conductivity, presenting a promising new direction for analog computing in deep learning applications. [6]

B. Distance Computation

Existing work proposes architectures for distance computation [7]. Calculating the distance between two vectors is a fundamental calculation commonly performed in Machine Learning for grouping or classification. There are several equations for calculating distance, such as the Euclidean distance, given by:

$$d_{\mathrm{E}}(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \tag{5}$$

where $\mathbf{x}$ and $\mathbf{y}$ are two points (vectors) in Euclidean n-space, and x_i, y_i are the coordinates of $\mathbf{x}$ and $\mathbf{y}$ respectively.

The Manhattan Distance, given by:

$$d_{\mathrm{M}}(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{n} |x_i - y_i| \tag{6}$$

The Mahalanobis distance, which considers the correlation between variables and is scale-invariant, is given as:

$$d_{\mathrm{Mh}}(\mathbf{x}, \mathbf{y}) = \sqrt{(\mathbf{x} - \mathbf{y})^T \mathbf{S}^{-1} (\mathbf{x} - \mathbf{y})} \tag{7}$$

where $\mathbf{S}^{-1}$ is the inverse of the covariance matrix of the dataset.

Proposed by [7], a brilliant analog architecture for solving these equations is displayed in Figure 1.

The circuit illustrated in Figure 1 computes all three of the previous distance formulas using configurable current-mode

[3]An imperative quality, discussed in [2]

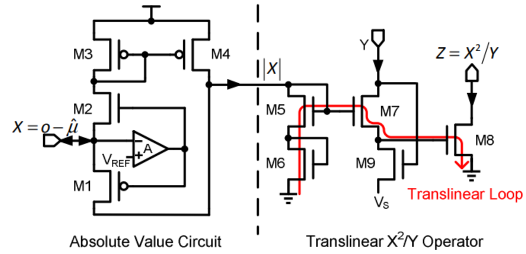

Fig. 1. Illustration of analog inference circuits from [7].

circuits. This study concluded that their system implementation demonstrated accuracies comparable to floating-point software but consumed orders of magnitude less power than a digital implementation.

C. Matrix Multiplication

With analog computing, we can reduce how often data has to be moved through memory; instead, we can perform calculations in place. Utilizing arrays of nonvolatile memory (NVM), matrix multiplication can be performed in $O(1)$ rather than with a combination of multiplication and summation operations, thereby lowering latency and power consumption [2]. The ability to perform these operations in constant time facilitates quicker convergence and inference for DNNs. Analog computations for matrix operations map 2-D matrices into physical arrays with the same number of rows and columns as the matrix. At the intersection of each row and column is an element with conductance G representing the strength between that row and column. By applying a voltage difference V across a given row and column pair, there is a current flow j.

$$j = GV \tag{8}$$

To generalize this concept, the components of a voltage vector $v_i (i = 1, ..., n)$ and collect the current at the n columns $j_i (j = 1, ..., m)$. Applying Ohm's and Kirchoff's law covers the current vectors to voltage vectors. By mapping the physical array with its connections g_{ij} to the matrix, the following equation is exactly equivalent to a matrix multiplication [2].

$$j_j = \sum_{i=1}^{n} g_{i,j} V_i \tag{9}$$

V. Conclusion

This review captures the newfound interest in and applications of analog computing in deep learning. Despite the natural strengths of an analog architecture, such as its natural capability for continuous data processing, energy efficiency, and speed, the path toward mainstream adoption of analog computing is not straightforward. There are still unresolved issues of programmability, scalability, and integration with existing digital infrastructures. Further research should be done into developing hybrid systems, which could ease the transition out of purely digital computing. In conclusion, despite legitimate challenges, the potential benefits of analog computing, when applied to deep learning, are too substantial to dismiss. Analog computing addresses the largest limitations in current Artificial Intelligence and Machine Learning development.

ACKNOWLEDGEMENTS

I want to express gratitude to D. Bryan Neel for the inspiration to write about Analog Computing. His passion for the topic was greatly encouraging in developing this work.

REFERENCES

[1] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016. [Online]. Available: http://www.deeplearningbook.org

[2] W. Haensch, T. Gokmen, and R. Puri, "The next generation of deep learning hardware: Analog computing," vol. 107, no. 1, 2019, pp. 108–122.

[3] S. Garg, J. Lou, A. Jain, Z. Guo, B. J. Shastri, and M. Nahmias, "Dynamic precision analog computing for neural networks," vol. 29, no. 2: Optical Computing, 2023, pp. 1–12.

[4] H. Schmid, "Combined analog/digital computing elements," in *Papers Presented at the May 9-11, 1961, Western Joint IRE-AIEE-ACM Computer Conference*, ser. IRE-AIEE-ACM '61 (Western). New York, NY, USA: Association for Computing Machinery, 1961, p. 299–314. [Online]. Available: https://doi.org/10.1145/1460690.1460725

[5] J. Schemmel, S. Billaudelle, P. Dauer, and J. Weis, "Accelerated analog neuromorphic computing," 2020.

[6] M. Onen, N. Emond, B. Wang, D. Zhang, F. M. Ross, J. Li, B. Yildiz, and J. A. del Alamo, "Nanosecond protonic programmable resistors for analog deep learning," vol. 377, no. 6605, 2022, pp. 539–543. [Online]. Available: https://www.science.org/doi/abs/10.1126/science.abp8064

[7] J. Holleman, I. Arel, S. Young, and J. Lu, "Analog inference circuits for deep learning," in *2015 IEEE Biomedical Circuits and Systems Conference (BioCAS)*, 2015, pp. 1–4.