# Memresisitive LSTM Architectures

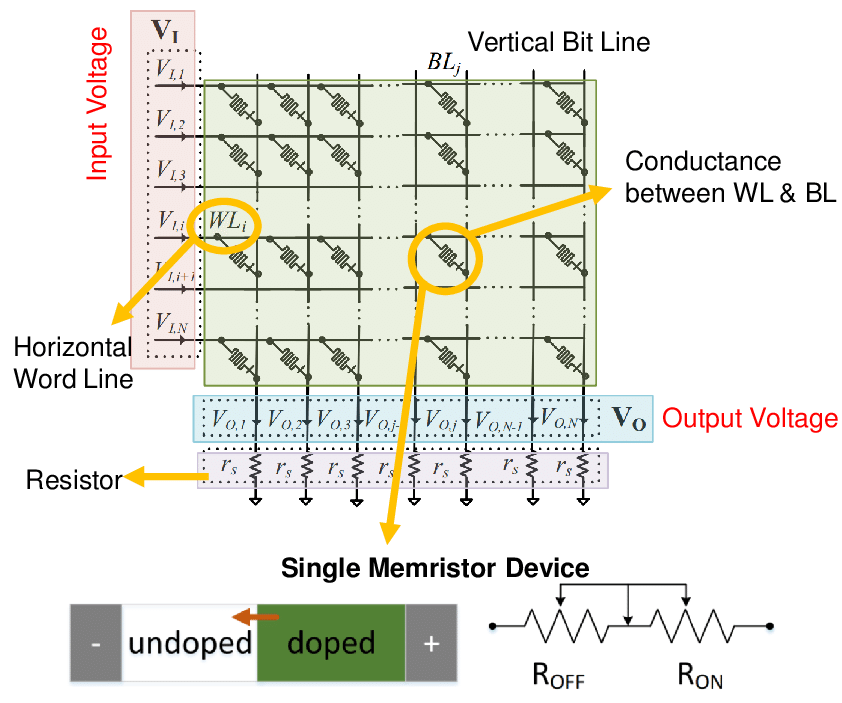
## Background

Memristor crossbar arrays carry out multiply-add operations in parallel in the analog domain, and so can enable neuromorphic systems with high throughput at low energy and area consumption. On-chip training of these systems have the significant advantage of being able to get around device variability and faults.

Diagram

Description automatically generated

Memristors represent a class of two terminal resistive switching multi-state memory devices. Research is around ensuring the validity of deep neural networks (like LSTM). The main benefit is the non-volatility property (ability to switch and store the resistive state even when power removed).



## Motivation and background

Kazybek Adam, Kamilya Smagulova and Alex Pappachen James review existing LSTM architectures used in Tensorflow (Coupled Inout and Forget Gate aka Gated Recurrent Networks).

The authors challenge the current approach of LSTM and present a memresitive LSTM architecture implementation in analog hardware. They propose a new central theme in the standard implementation of LSTM and note other architecture variations can be constructed by rearranging, adding or deleting analog circuit parts and adding extra crossbar rows.

## Research objective/question(s)

Kazybek Adam, Kamilya Smagulova and Alex Pappachen James transition from equations to the hardware implementation of LSTM. The main objective is efficiency in the neural network as implemented purely analog.

The authors propose a hardware implementation and simulate a time series forecasting problem (changing of the number of international airline passengers). The authors go into each of the building blocks of building a memresitive LSTM hardware implementation and then apply these building blocks into a proposed circuit design of these building blocks applied to this time series problem.

----

Alvesson and Sandberg’s study integrates both positive and negative research agenda by developing a novel methodology for identifying and challenge assumptions in existing theories. First, the objective is to develop a typology of what types of assumptions can be challenged. Second, to establish a set of methodological principles on how this can be accomplished.

The authors challenge the view that what makes a theory influential is truth. When a theory challenges the assumptions in a significant way it is said to be seen as meaningful. Therefore, can we leverage a new method to identify and challenge assumptions and with that understanding then lead to more influential theories. First the authors question what types of assumptions are relevant to consider in existing literature and second they ask how these assumptions can be identified, articulated and challenged such that they lead to new interesting theories. To develop the methodology the authors, answer two questions

## Prior relevant work/literature gap

Smagulova (2018) provided a purely analog implementation, however, was limited to the individual building blocks. For example, analog circuits were provided for the activation function, and element-wise multiplication operations. The original research also presented a possible implementation of a memresitive crossbar circuit.

## Theory, conceptual framing

The full potential of nuero-memrisitive systems is yet to be realized; however, the belief is they will provide new ways to develop higher levels of machine cognition. The possibility to scale the implementation of high-density large scale neural networks in integrated chips offer several advantages including (lower power, near-sensor analog computation over CMOS). Dot-product computations using a crossbar array can be mapped directly into the neural network layers.

This leads to opportunities for data driven (e.g IoT) and integrated into wearable, mobile, edge and computing devices. Implementation of memristor based neural computing is area and power efficient.

## Methods

Memristive Standard LSTM using crossbar circuits represent the weights in the neural network. The rest of the circuit is designed for the addition, multiplication, and activation functions.

Leveraging Smagulova (2018) the authors combine the separate building blocks into a system. Further the authors expand on current-based memristive LSTM from Smagulova et al (2018) in the design of CMOS which is a circuit designed to be used in a sequential manner (meaning it needs to run through M cycles to obtain the hidden unit vector and cell state vector). This became the base of the authors experience.

The authors also note other methods including simulation of networks in software C++ in Gokmen et al (2018) and a Memristive LSTM chip that was fabricated in prior research LI C et al (2018) however this one also had elements of the operations built in software.

To fill the gaps across these (4) methods the authors implement a fully functional and purely analog circuit simulation combining these methods and applying it a time-series prediction. This voltage-based system was divided into three parts: Vector matrix Multiplication, multiplication circuits and activation function.

## Results

The main contribution of this paper is development of the circuit and simulation performance results obtained from software and hardware (analog).

An interesting observation is the activation function (both activation, sigmoid and hyperbolic tangent) used in LSTM are all implemented with the same circuit (they noted the similar waveforms when plotted). Vector matrix multiplication is noted as the benefit of memristor crossbar circuits as handling the operation efficiently. In order to solve a positive only synapse two memristor per synapse are added to implement both positive and negative weights.

The training set was converted into three column-form with the first two columns as input and third column as target. The circuit implementation was a simple two-layer neural network

Diagram

Description automatically generated

Chart, histogram

Description automatically generated

* MSE and RMSE of software models are 0.0113 and 0.1059
* MSE and RMSE of hardware models are 0.0101 and 0.1004

Total circuit time (for predicting 45 test points) was 3.96ms (with 2ms ; the software version was not noted in the paper.

This paper introduced a voltage-based LSTM circuit design for predicting the number of airline passengers and produced similar results closely matching MSE and RMSE. By eliminating the software component and eliminating converting from currents and voltage expected efficiency should reduce runtime complexity; however, the authors do not provide runtime differences.

## Discussion

The paper provides compelling evidence that a hardware implementation can provide results close to the software simulation results and notes that simulation times can be further reduced through control voltage signals by making sure there are not convergence issues (the authors implemented a 2ms intentional delay to prevent convergence issues).

LSTM using analog CMOS-memristive circuits was validated for time-series prediction applied to predicting airline passengers.

## Thoughts

The authors surmise that LSTM neural networks in circuits would demonstrate higher accuracy and with the newer memristive circuits would provide a more efficient method for computationally expensive dot-product and weight-matrix multiplication within an LSTM cell. However, although efficient the method employed was sequential and requiring self-imposed interrupts to solve convergence issues (impedance issue in the circuit). Definitely a lot more work to do in the area but along the lines of CPU to GPU to TPUs, advanced neural circuits are getting much more traction and I expect that to continue.