Deep Neural Networks for Behavioral Modeling of Analog ICs

Introduction to the Research in Electrical and Computer Engineering

André Carneiro Amaral

Instituições Associadas

TÉCNICO LISBOA

universidade de aveiro

UNIVERSIDADE D COIMBRA

altice labs

NOKIA

UNIVERSIDADE BEIRA INTERIOR

U.PORTO

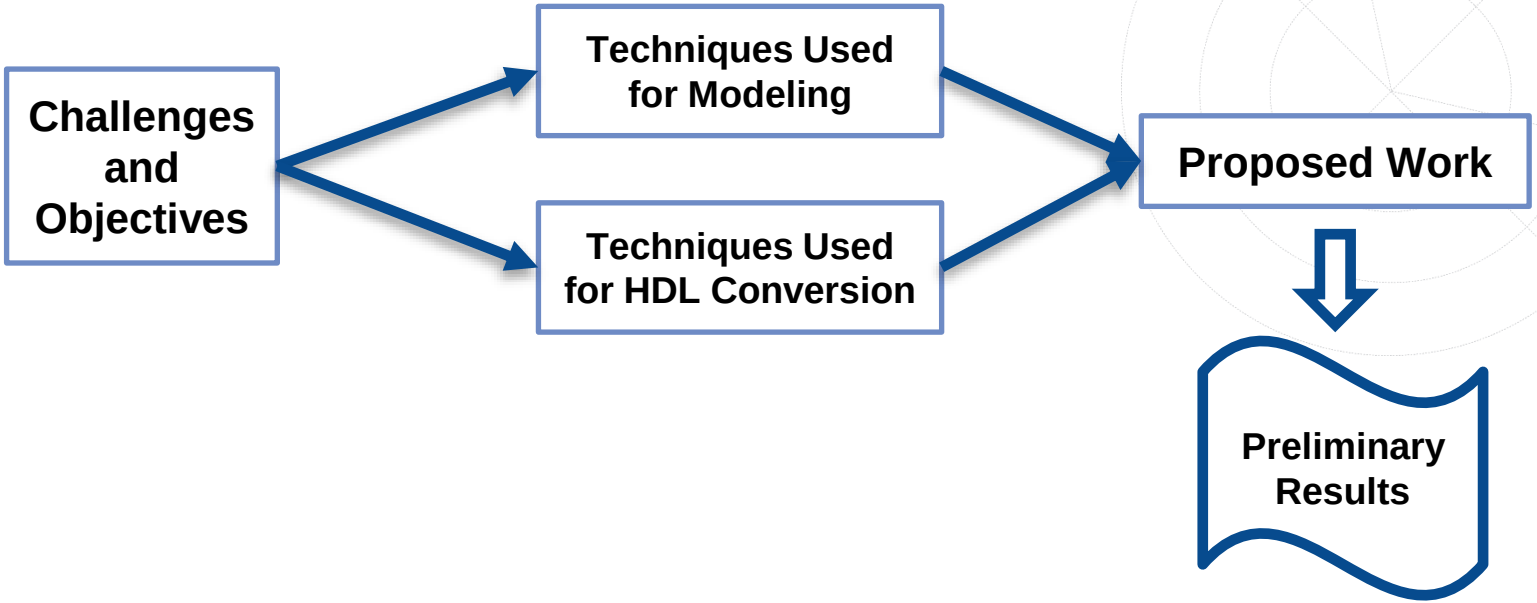
ISCTE IUL Instituto Universitário de Lisboa

IPL instituto politécnico de leiria

Instituto Superior Técnico, February 2022

instituto de telecomunicações

Introduction



Challenges and Objectives (1)

Difficulties

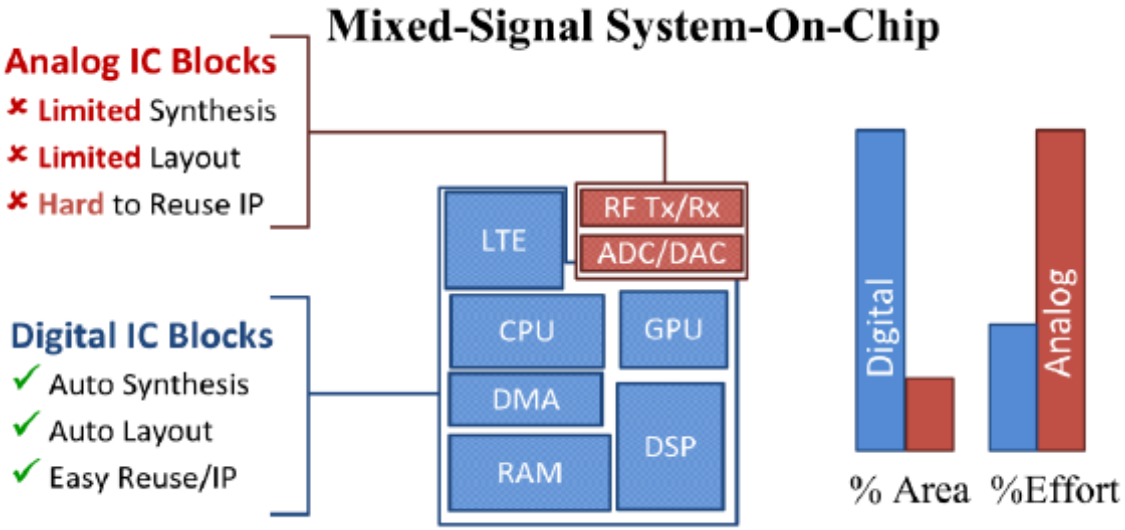


Figure 1 – MS SoC Design [1]

Challenges and Objectives (2)

Importance

Why Analog Circuits Are Important?

Figure 2 – Analog Sensors

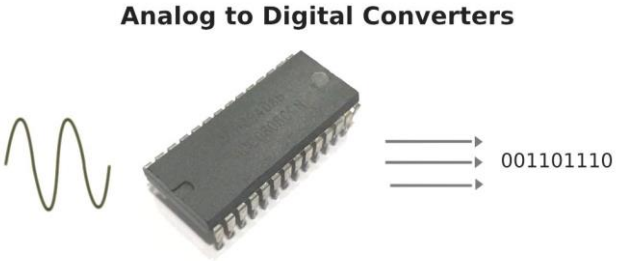


Figure 3 – Analog to Digital Converter

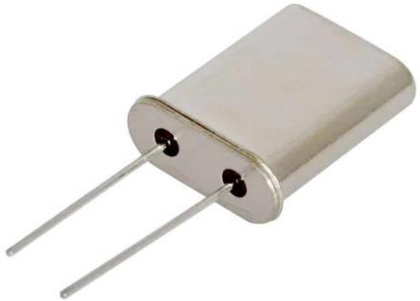


Figure 4 – 200MHz Cristal Oscillator

Challenges and Objectives (3)

Objectives

1) Create a model that capture the circuit behavior using Machine Learning Techniques
2) Speed-Up the simulations

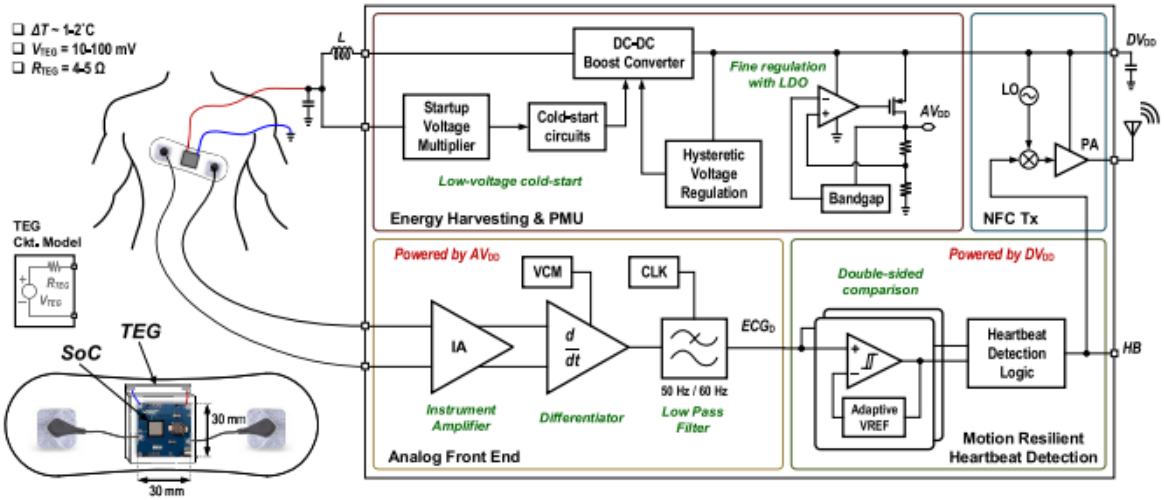


Figure 6 – Motion-resilient heartbeat detection SoC architecture [3]

Figure 5 – AIDAsoft [2]

Challenges and Objectives (4)

Proposed Work

First lets see what already exists!

State-of-Art (1)

Charge Pump Circuits [4]

Results

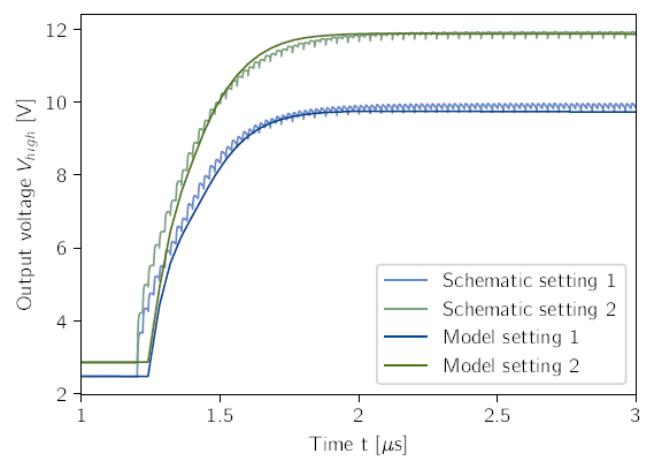


Figure 9 – Model Behavior

TABLE I
SIMULATED PARAMETER SETS FOR TRAINING AND VALIDATION DATA.

Parameter	Training	Validation
V_{dd}	lin [1.2 : 0.6 : 3.6] V	lin [1.2 : 0.3 : 3.6] V
R_{load}	log [100k : 10 : 1M] Ω	log [100k : 20 : 1M] Ω
f_{clk}	lin [10M : 10M : 50M] Hz	lin [10M : 5M : 50M] Hz
N	250	1620

Figure 10 – Model Results

State-of-Art (2)

Circuit's Power Consumption [5]

Results

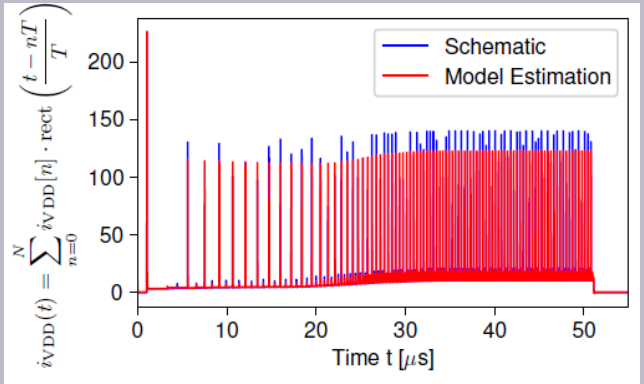


Figure 13: Transient Supply Current

TABLE I
COMPARISON BETWEEN DIFFERENT REPRESENTATIONS OF THE OSCILLATOR BLOCK.

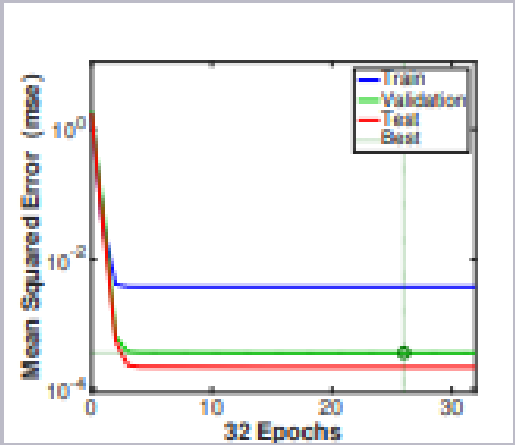
	Simulation Time	Energy Consumption in Interval t=[0 ,55 µs]
Transistor Level	2 min 48 s	623 pJ
Functional Model	0 min 7 s	0 pJ
Augmented Functional Model	0 min 12 s	640 pJ

Figure 14: Model Results

State-of-Art (4)

Multiple Inputs Multiple Outputs [7]

Results



Graph of MSE

TABLE II

TRANSIENT SIMULATIONS PERFORMED FOR THE TRAINING DATA COLLECTION PURPOSES

Simulation	Simulation Time	CPU time	Input	Output	♯ of samples
Trimming	100 μs	1.4 s	Trimming inputs	$Vout_{1V}$	695
Load Jump	540 μs	1.3 s	Load Profile	$Vout_{1V}$	433
Line Jump	200 μs	1 s	V_{DD}	$Vout_{1V}$	501

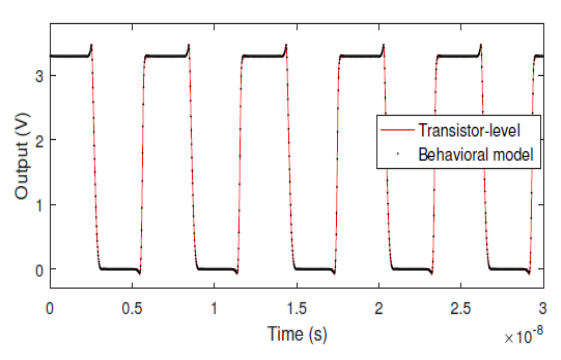
Model Results

State-of-Art (5)

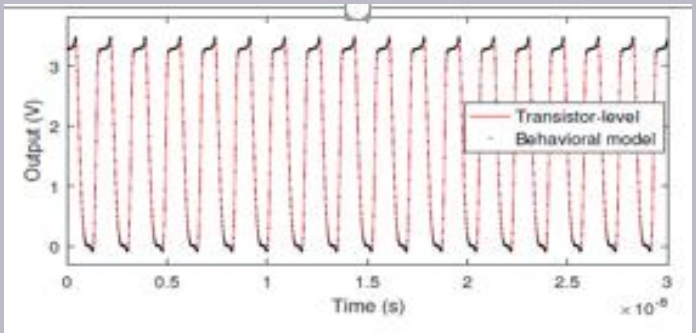
Simple Oscillator [8]

Results

1) K_osc = 0.1698 Inverter Ring Oscillator.



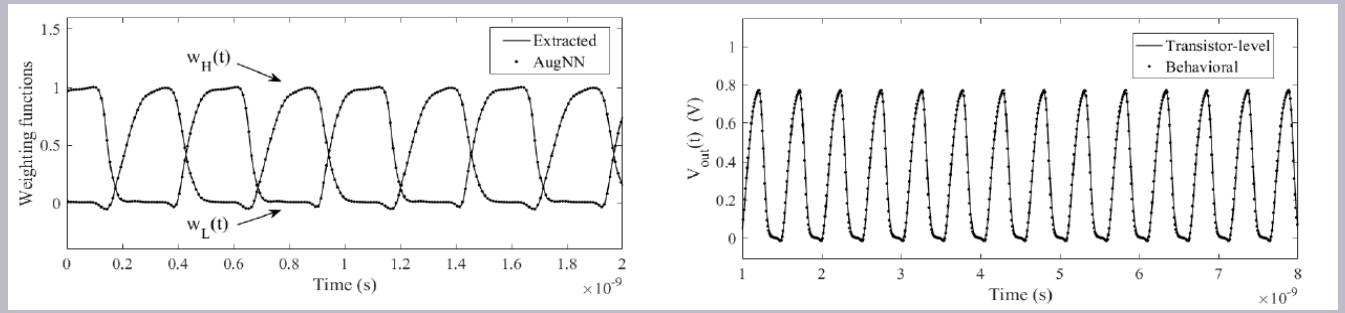
2) K_osc = 0.5734 Inverter Ring Oscillator.



State-of-Art (6)
Oscillator with Buffer [9]

Results

K_osc = 1.95GHz Ring Oscillator with a Buffer.



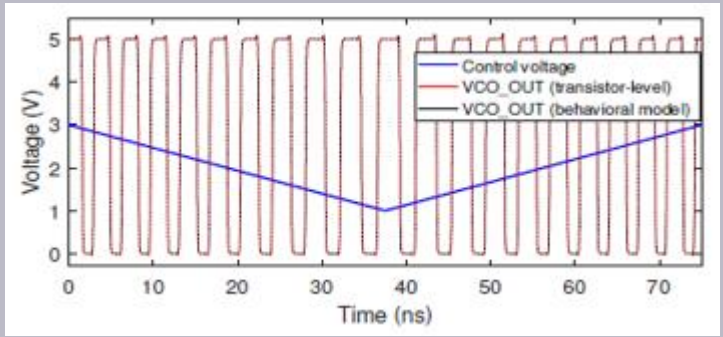
State-of-Art (7)

Oscillator with Voltage Controller [10]

Results

Voltage Controller Oscillator (VCO)

[1V, 3V] – Range of voltage controller sweeping

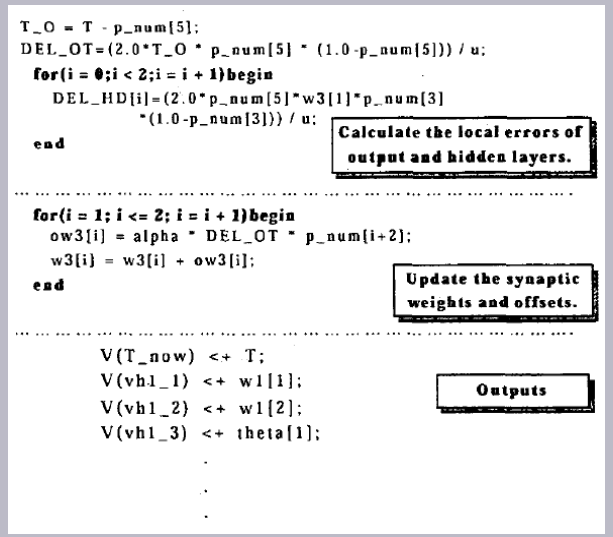


Ref.	Device/Circuit	Method(s)	Objective	HDL	Error (circuit vs model)	Time (time reduction)	Data (simulator / dataset [in], [out])
[4]	RF- microwave components and MESFET	ANN (several)	Review of ANN based CAD for microwave designs.	-	-	-	Spectre / $[v_{gs}, V_{ds}, f]$, [Drain Current]
[7]	Three-stage cross-coupled charge pump	ANN (MLP)	Model a charge Pump circuits	Verilog-AMS	4.85%	7s	Spice / $[v_{dd}, R_{load}, f_{clk}]$, [Output Voltage]
[10]	Low Relaxation Oscillator	ANN (TDNN)	Model to analyze the Power consumption	SystemVerilog	2.70%	2min	Spectre / $[f, clk, EN]$, [Supply Current]
[11]	Analog-n/d	ANN (Bprop)	Modeling the power consumption.	Verilog-XL	1.53%	-	- / $[f, V_{out}]$, [Power]
[12]	RF Receiver, Sigma Delta, Low-Dropout Regulator	ANN (RNN) + Genetic Algorithm	Surrogate Model with Collaborative Stimulus	Verilog-A	RMSE: $31\mu V$	30* Faster	- / [Stimulus], [Adapted Stimulus]
[13]	CMOS band-gap voltage reference circuit (BGR)	CompNN (NARX + TDNN)	Capture the dynamic of analog MIMO circuits	Verilog-A	5.00%	17* Faster	- / [Trimming, Load Jump, Line Jump], [1V-Out]
[15]	Inverter Ring Oscillator	ANN (FFNN)	Capture the periodicity of oscillator output waveform	Verilog-A	-	10* Faster	Spectre / [Time Step, f_{osc}], [Output Voltage]
[16]	Ring Oscillator with a Buffer	ANN (FFNN + RNN)	Capture the periodicity of the oscillator + buffer output waveform	Verilog-A	RMSE: 0.0034	10* Faster	Spectre / [Time Step, f_{osc}], [Output Voltage]
[17]	Transistor-Level VCO Circuit	ANN (2 FFNN + RNN)	Model the oscillatory frequency and see the effects on the output waveform.	Verilog-A	RMSE: 0.0061	10* Faster	Spectre / [Time Step, $f_{osc}, V_{controller}$], [Output Voltage]
[21]	SCFL Buffer Cell, Resistive Mixer, Voltage Controller Oscillator	SVM ($\varepsilon - SV\ regression$)	Robust modeling of GaAs transistors and circuits.	-	MSE: $3*10^{-7}$	-	- / [V(n), I(n-k)], [Output Current]
[22]	Analog Circuits- CMOS	SVM	Efficient active learning scheme for feasible design space selection	-	-	-	Spice / [Heigth, Width]

State-of-Art (8)

Verilog and ANN [11]

Results (4)



- Calculates the error of the output and the hidden layers.
- The weights are updated and the outputs are formulated.

Reference	Neural Network Type	VHD Language
[23]	Backpropagation	Verilog-A
[24]	RNN	Verilog-A

Proposed Work (1)

Why use Artificial Neural Networks?

Tribe	Origins	Master Algorithm	Examples of Methods
Symbolist	Logic	Inverse Deduction	Decision Trees
Connectionist	Neuroscience	Backpropagation	Deep Neural Networks, Perceptron
Evolutionaries	Evolutionary Biology	Genetic Programming	Genetic Algorithm
Bayesians	Statistics	Probabilistic Inference	Naïve-Bayes, LDA, PCA
Analogizers	Psychology	Kernel Machines	SVM, KNN

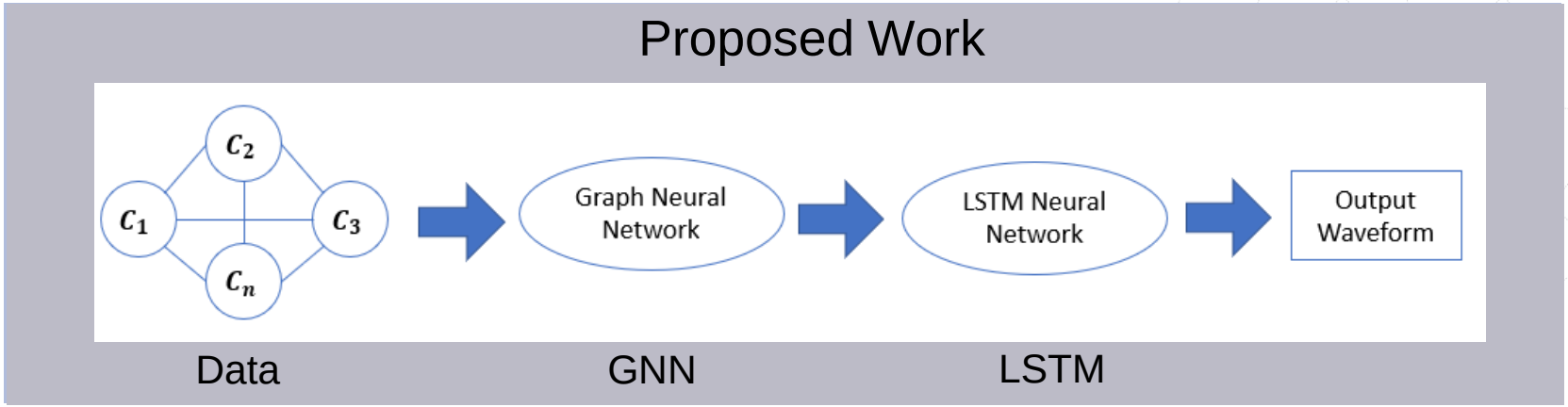
Baysians & Analogizers – Commonly used for classification problems (not only!).

Symbolist – In case of a lot of constraints is hard to produce an efficient solution.

Evolutionary – High simulation time

Proposed Work (2)

Proposed Workflow



Proposed Work (3)

Data Acquisition and Organization (1)

- Data Acquired through SPICE Simulations;

1. The input parameters are sweep to obtain better representation.
2. The output waveform is extracted for each set of inputs.

- Using a set of different Amplifiers topologies;

1. Widley studied, providing a good baseline to evaluate the results.
2. Readily available in ICG-Lx group, and were used in AIDA and DeepPlacer tools.
3. The implemented techniques can be easily integrated into larger systems.

Proposed Work (4)

Data Acquisition and Organization (2)

- Data Organized in a Graph;

1. Nodes represent the devices.
2. Edges represent the number of wires that connect devices.

} Adjacency Matrix

- Data Curation;

1. Remove null values.
2. Filter the adjacency matrix.
3. Normalize the adjacency matrix rows.
4. Normalize the input features.

Proposed Work (5)

Graph Neural Network

Some math….

GCN Layer: $H^{L+1} = \sigma[(\hat{D}^{-\frac{1}{2}}A\hat{D}^{-\frac{1}{2}})H^{L}W^{L}]$

A -> Adjacency Matrix
$\hat{D}$ -> Diagonal Node Degree Matrix
H^{L} -> Previous Hidden Layer
H^{L+1} -> Next Hidden Layer

Proposed Work (6)

"Long-Short Term Memory" Neural Network (LSTM)

Problems to Avoid!

Vanishing Gradients

The gradient can decrease exponentially if the derivatives are very small.

How to avoid

- Use ReLU
- Batch Normalization
- Different weights initializations (Xavier Initialization)

Exploding Gradients

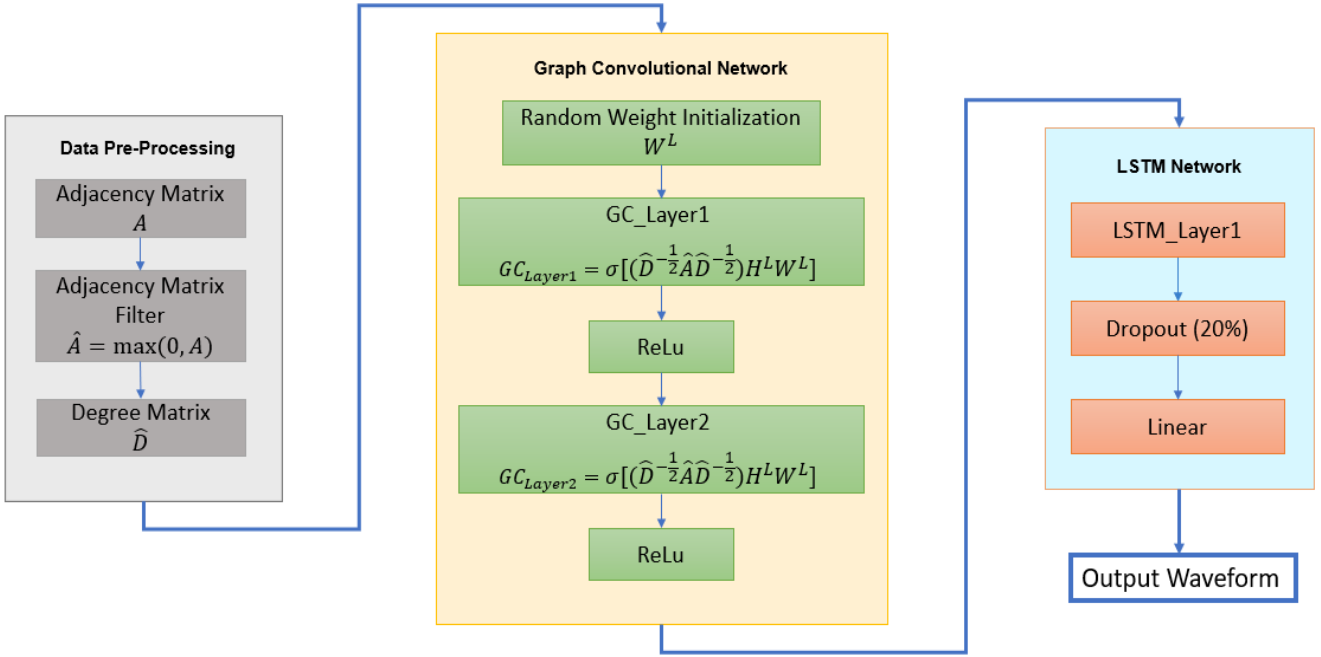
The gradient can increase exponentially if the derivatives are very large.

How to avoid

- Use Gradient Cliping:
  → Norm Clipping
  → Elementwise Clipping
- Batch Normalization

Proposed Work (7)

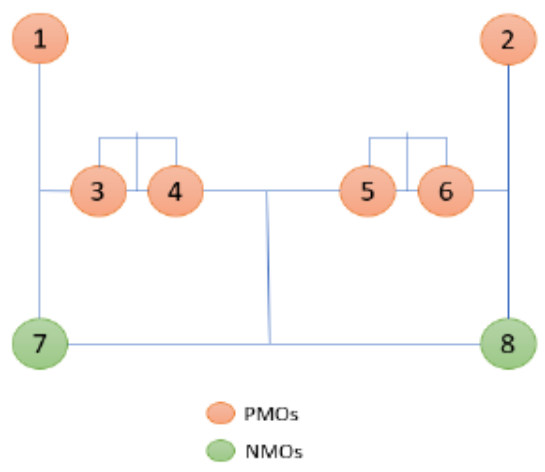
Proposed Workflow Detailed



Preliminary Results (1)

Graph and Adjacency Matrix

Adjacency Matrix Construction



	1	2	3	4	5	6	7	8
1	0	0	1	0	0	0	1	0
2	0	0	0	0	0	1	0	1
3	1	0	1	1	0	0	1	0
4	0	0	1	1	1	0	1	1
5	0	0	0	1	1	1	1	1
6	0	1	0	0	1	1	0	1
7	1	0	1	1	1	0	0	1
8	0	1	0	1	1	1	1	0

Each node is linked to what nodes?

Preliminary Results (2)

Dataset

Data Curation Code

```
training_set_param = df.drop(['/out- Y', '/out+ Y'], axis = 1).values
training_set_labels = df[['/out- Y', '/out+ Y']].values
```

Features and Labels

```
def sliding_windows(data1, data2, seq_length):
    x = []
    y = []

    for i in range(len(data1)-seq_length-1):
        _x = data1[i:(i+seq_length)] # input: the seq_length of data
        _y = data2[i+seq_length] #Output: the next value after the input
        x.append(_x)
        y.append(_y)

    return np.array(x),np.array(y)
```

Sliding Window

```
scaler = MinMaxScaler()
training_set_param = scaler.fit_transform(training_set_param)
```

Normalization

Preliminary Results (3)

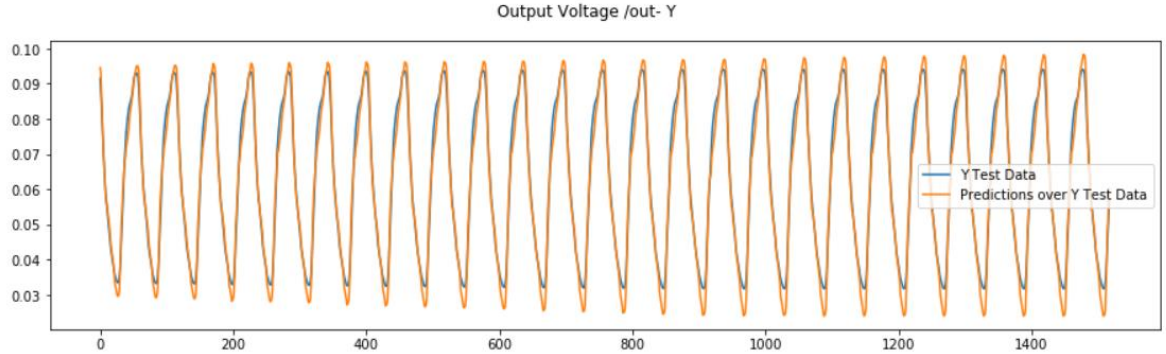
LSTM Model

Training Phase

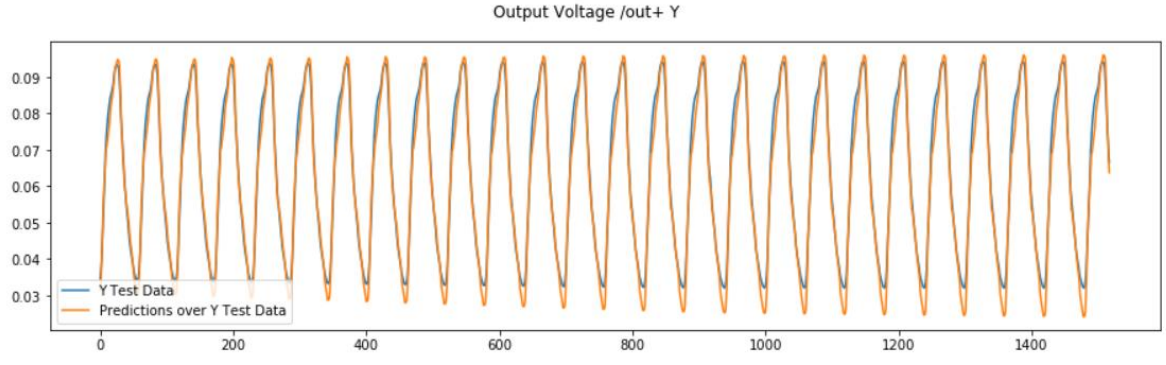
```
number_of_epochs = 500

for epoch in range(number_of_epochs):
    # Initialize the gradient to avoid value agregations
    optimizer.zero_grad()

    # Predict the model using the training set
    y_pred = lstm(trainX)

    # Obtain the loss
    loss = criterion(y_pred, trainY)

    # Backpropagation
    loss.backward()
    optimizer.step()
```

Preliminary Results (4)

Results Obtained



$$MSE = 1.2710 * 10^{-5}$$



$$MSE = 1.2706 * 10^{-5}$$

References

[1] - Martins, R., Horta, N. and Lourenço, N., “Automatic Analog IC Sizing and Optimization Constrained” (1st ed.), Springer, 2017

[2] - https://www.aidasoft.com/Home

[3] - Bose S., Shen B., Johnstion M. (2020). “A Batteryless Motion-Adaptive Heartbeat Detection System-on-Chip Powered by Human Boady Heat”. IEEE Journal of Solid-State Circuits, Vol. 55, No.11.

[4] - Grabmann, M., Landrock, C. and Gläser, G. (2021), “Machine Learning in Charge: Automated Behavioral Modeling of Charge Pump Circuits”. SMACD / PRIME 2021.

[5] - Grabmann, M., Landrock, C. and Gläser, G. (2019), “Power to the Model: Generating Energy-Aware Mixed-Signal Models using Machine Learning”. SMACD 2019, Lausanne, Switzerl.

[6] - Lei, J.Y and Chatterjee, A. (2021), “Automatic Surrogate Model Generation and Debugging of Analog/Mixed-Signal Designs Via Collaborative Stimulus Generation and Machine Learning”. 26th Asia and South Pacific Design Automation Conference.

[7] - Hasani, R. M., Haerle, D., Baumgartner, C. F., Lomuscio, A. R. and Grosu, R. (2017). “Compositional neural-network modeling of complex analog circuits”. International Joint Conference on Neural Networks.

[8] - Yu, H., Swaminathan, M., Ji, C. and White, D. (2017). “A method for creating behavioral models of oscillators using augmented neural networks”. Conference on Electrical Performance of Electronic Packaging and Systems.

[9] - Yu, H., Swaminathan, M., Ji, C. and White, D. (2018). “Behavioral Modeling of Steady-State Oscillators with Buffers Using Neural Networks”. Conference on Electrical Performance of Electronic Packaging and Systems.

[10] - Yu, H., Swaminathan, M., Ji, C. and White, D. (2018). “A Nonlinear Behavioral Modeling Approach for Voltage-controlled Oscillators Using Augmented Neural Networks”. IEEE/MTT-S International Microwave Symposium - IMS.

[11] - Suzuki K., Nishio A., Kamo A., Watanabe T. and Asai H., (2000). “An application of Verilog-A to modelling of back propagation algorithm in neural networks”. 43rd IEEE Midwest Symposium on Circuits and Systems.