MP-36

MINI PROJECT PRESENTATION

Kushagra Agarwal

Aryamaan Jain

ABSTRACT

Through Simulation softwares, the Delay and Leakage values for a Full Adder circuit of MOSFETs were computed for 50000 different Length and Width parameters for all the 28 transistors. Using these values as training data, we created a Neural Net Model implemented on the Keras framework to predict the delay and leakage values for unknown L and W parameters. Then we applied Genetic Algorithms and Particle Swarm Optimisation techniques to find the best suitable 56 values (28 L and 28W) for which the Delay and Leakage values were the least.

MOSFETs Introduction

FETs have a few disadvantages like high drain resistance, moderate input impedance and slower operation. To overcome these disadvantages, the MOSFET which is an advanced FET was invented. MOSFET stands for Metal Oxide Silicon Field Effect Transistor or Metal Oxide Semiconductor Field Effect Transistor. This is also called IGFET meaning Insulated Gate Field Effect Transistor. The FET is operated in both depletion and enhancement modes of operation. The following figures show how a MOSFET looks.

The construction of a MOSFET is a bit similar to the FET. An oxide layer is deposited on the substrate to which the gate terminal is connected. This oxide layer acts as an insulator (SiO_2 insulates from the substrate), and hence the MOSFET has another name as IGFET. In the construction of MOSFET, a lightly doped substrate is diffused with a heavily doped region. Depending upon the substrate used, they are called as P-type and N-type MOSFETs.

What motivated the enquiry

- The enquiry was motivated because of non optimum power consumption by electrical circuits like adders because optimal width and length sizes are not known.
- Growing field of artificial intelligence with the help of neural networks helped us to first build a model and then optimize using algorithms such as genetic algorithms and particle swarm optimization.

Question trying to answer

- We tried to answer and find the best and most efficient methods to optimize given circuits.
- We were able to see that methods such as linear regression were not good and scalable and hence shifted to methods such as neural networks using Keras.
- We were able to optimize using algorithms such as genetic algorithms and particle swarm optimization algorithms.

Methods

Our solution was divided into two parts. The first one was to build a good model. We used neural networks to build a model after the initial failure to do so by using linear regression. We were not able to use linear regression because of its huge memory consumption. Neural network did not suffer from this problem and was able to give a good R2 score of 0.99 whereas linear regression gave a score of only about 0.75. After building a model we had to optimise the model. We optimised our model using two algorithms. The first one was genetic algorithms and the second one particle swarm optimisation. Both algorithms were able to converge on delay but were not able to do very well on the leakages. All this was done using popular libraries and frameworks like Keras, Scikit, Pandas, etc.

Materials

1. Keras for model building
2. Scikit learn for model evaluation
3. Pandas for data handling
4. Microsoft teams for communication
5. Outlook for communication

Procedure

We built our neural network model using Keras with Tensorflow backend. To build the Neural Net, Deep Learning was used. The Neural consisted of 9 layers. The input layer had 56 nodes. The hidden layers had 43, 35, 29, 25, 20, 26, and 12 nodes while the output layer had 6 nodes for the delay model and 1 node for the leakage model. All the layers but the last one used ReLu activation function and the last layer used a linear regressor alone. Adam optimization technique was used to converge faster. The two models were trained independently on data consisting of 40000 rows for delay and leakage values separately. Both the models were tested using both R2 scores and MSE to get an idea of their performance on an unseen test set of 10000 rows. We optimised our model parameters given initial sizing using genetic algorithms and particle swarm optimization. To do this the cost function was defined such that it took into account both the delays and leakages. Hence both Neural nets were called simultaneously to provide the values for a test set of 56 parameters. Then the results were compared to initial sizing to find out the efficiency of the framework.

Results

Genetic Algorithms

optimised delays =

[4.8374468e-12 8.0630624e-12
8.1829803e-12 1.4586535e-11
9.5415099e-12 1.1679768e-11]

optimised leakage = 1.5381e-06

Particle Swarm Optimisation

optimised delays =

[5.5920e-12, 1.1030e-11,
5.4595e-12, 1.3326e-11, 5.7792e-12,
9.7994e-12]

optimised leakage = 1.3509002e-06

Discussion

- This mini project helped us to know about various techniques used in circuit optimization.
- We enjoyed our collaboration with Dr. Zia Abbas and his research students.
- We were able to apply many of the topics we had previously learned in courses such as Machine, Data and Learning to a real world problem and solve it to some extent.
- Now we will be able to solve any problems of similar structure given to us using the framework and ideas of this mini project.

Conclusion

- We were able to optimize our given problem to some extent.
- We learnt new tricks and techniques which can be used further on in future work.
- We achieved a model of R2 score 0.99 and were able to converge our model for delays but not so much for leakages.
- Both Genetic and Particle Swarm Optimisation Technique were able to converge. The delay values reduced considerably by 34% as compared to the initial sizing and the leakage went up by only 8%. The inverse relation between the two parameters resulted in such a behaviour nevertheless this skewness of 34 versus 8 gives weight to the credibility of our proposed framework.

Recommendations for further research

The accuracy of the Neural Net could be further improved by building a deeper net or deploying different Machine Learning Architectures.

Different optimization algorithms apart from GA and PSO could be used for the convergence of the framework like Glow worm Optimisation.

The hyperparameters used for PSO and Genetic Algorithms can be tuned further like the social parameters a, b and w in PSO and the mutation probability etc in GA.

The cost function could be changed to achieve a different descent if the model is suspected to be stuck in a suboptimal local minima.

Citations and Bibliography

1. Group method of data handling (GMDH): Alexey Ivakhnenko and Lapa

2. Semi-Supervised Learning with Ladder Network: Antti Rasmus, Harri Valpola, Mikko Honkala, Mathias Berglund, Tapani Raiko

3. Adaptation in Natural and Artificial Systems: J.H. Holland

4. Genetic algorithms: concepts and applications [in engineering design]: K.F. Man ; K.S. Tang ; S. Kwong 7. Genetic algorithms for modelling and optimisation: JohnMcCall

5. Implementation of the PSO algorithm for solving the JSSP in parallel: Wael Abdel-Rehim

6. Particle Swarm Optimization: Kennedy, J.; Eberhart, R.