



**INDIAN INSTITUTE OF TECHNOLOGY GUWAHATI  
SHORT ABSTRACT OF THESIS**

Name of the Student : Rituparna Choudhury

Roll Number : 176102101

Programme of Study : Ph.D.

Thesis Title: VLSI Implementation of Training Accelerators for Decision Tree Algorithm

Name of Thesis Supervisor(s) : Prof. Shaik Rafi Ahamed and Dr. Prithwijit Guha

Thesis Submitted to the Department/ Center : EEE

Date of completion of Thesis Viva-Voce Exam : 24-07-2023

Key words for description of Thesis Work : Machine Learning, VLSI, FPGA, ASIC, Decision Tree, Verilog HDL, training accelerator.

---

**SHORT ABSTRACT**

This thesis presents the hardware realization of three DT algorithms namely, Two Means Decision Tree (TMDT), Hybrid Decision Tree (HDT), and Perceptron Decision Tree (PDT). The TMDT algorithm classifies the data based on two-means clustering in each node. This reduces the computation to a great extent and thus, results in efficient hardware. First, the offline implementation of TMDT training is performed where the entire data is loaded to on-chip memory on Field Programmable Gate Array (FPGA). In this work, the TMDT algorithm is implemented in serial and mixed mode for binary classification only. However, the serial processing of blocks limits the speed of execution. So, next, a mixed implementation of TMDT training is proposed on FPGA. In this work, some blocks are implemented in pipelined manner and some blocks in parallel to minimize the latency. The memory access latency increased to a great extent due to large on-chip memory. Also, the huge memory consumption limited the training data size. So, the TMDT is modified and a batch-mode training process is implemented for multi-class classification on FPGA in the next chapter. In this implementation, the training data is divided into batches and one batch at a time is loaded into the chip memory to train the DT. This batch-mode implementation removed any constraint on the training data size. The accuracy of TMDT can be enhanced by implementing hybrid nodes. So, next, HDT and its training implementation on FPGA are proposed. This DT is a hybrid combination of split nodes hosting mean-dependent and axis-aligned split decision functions. The mean-dependent nodes are similar to the TMDT nodes and the axis-aligned split function parameters are learned by reducing the number of impurity computations. The HDT is observed to perform better than TMDT. Although there is a little increase in training latency, the performance measures (accuracy and F1-score) were observed to improve. However, the HDT performance was still inferior for small-sized datasets. To further improve the accuracy of the small-sized datasets, PDT training is implemented on FPGA. In this DT, each split node hosts a single output perceptron (no hidden layer). For implementing a perceptron efficiently in hardware, the perceptron architecture consisting of Offset Binary Coding (OBC) and Co-Ordinate Rotation-axis Digital Computer (CORDIC) is proposed and implemented on both FPGA and Application Specific Integrated Circuits

(ASICs). The OBC architecture has been used in literature for inner product calculation in filters where multiplication is implemented using shifters and adders. Also, training hardware for PDT is proposed on FPGA. Next, classification hardware for PDT is proposed for bio-medical applications on both FPGA and ASIC. The DT algorithms presented in this thesis have lower complexity compared to CART, C4.5 or ID3 algorithms. The training of these DT algorithms is implemented on FPGA and found to be both resource-efficient than the existing training accelerators. The serial architecture consumes fewer resources and runs at least 10x faster than the software implementation. The serial implementation of TMDT optimizes the resource consumption by at-least 8x as compared to the existing training accelerator for CART. The mixed implementation is found to be at least 14x faster than the software implementation. The batch-mode implementation is found to speed up the training by at least 27x as compared to software implementation. It halves the LUT consumption as compared to previous designs. It also exhibits 10x reduction in BRAM usage as compared to the training accelerator for CART. The PDT training is accelerated by 34x for the worst-case scenario (largest dataset) as compared to software implementation. This design almost halves the resource utilization as compared to previous designs and has 6x saving in BRAM consumption as compared to CART training accelerator. This hardware achieves a speed-up by a factor of 2 as compared to the software.

