Branch Prediction Models Exploration

Daniela Trevina
Department of EECS
Texas A&M University-Kingsville
Kingsville, USA
daniela.trevino@students.tamuk.edu

Daniela Lopez
Department of EECS
Texas A&M University-Kingsville
Kingsville, USA
daniela.lopez@students.tamuk.edu

Mengxiang Jiang
Department of EECS
Texas A&M University-Kingsville
Kingsville, USA
mengxiang.jiang@students.tamuk.edu

Samah Allahyani
Department of EECS
Texas A&M University-Kingsville
Kingsville, USA
samah.allahyani@students.tamuk.edu

Ugochukwu Onyeakazi
Department of EECS
Texas A&M University-Kingsville
Kingsville, USA
ugochukwu.onyeakazi@students.tamuk.edu

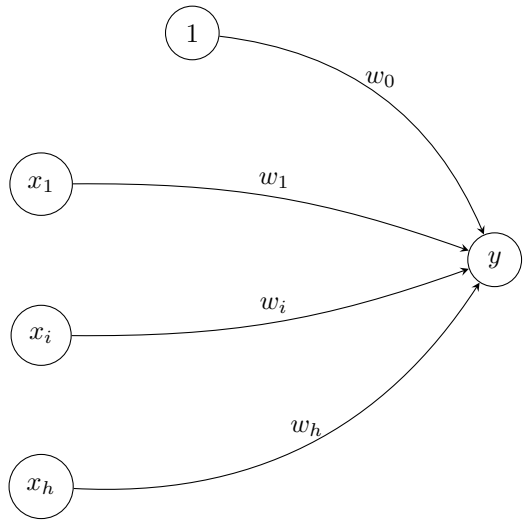

Fig. 1. Perceptron Predictor

***Abstract*—**

I. Introduction

II. Models

In this section, we will cover the various branch predictor models used in the simulation.

A. Perceptron Predictor

This model is a single-layer version of an artificial neural network that can identify and classify patterns, first applied to branch prediction by Jimenez and Lin [1]. In the model shown in Fig. 1 are the input vector (x), the weight vector (w), and the output (y). The inputs correspond to values taken from the global history register, with the exception of x_0, the bias, always set to 1. The output's sign determines the prediction. If it is negative, it is not taken, otherwise it is taken. Equation (1) shows the calculation of the output:

$$y = w_o + \sum_{i=1}^{h} x_i w_i \tag{1}$$

The weights of the perceptron are trained using the algorithm below:

```
if sgn(y) != t or abs(y) <= theta  then
    for i := 0 to n do
        w[i] := w[i] + t*x[i]
    end
end
```

t is 1 if the branch is actually taken or -1 if not. θ is the threshold to determine when training should stop. w_i is incremented if t and x_i agree, and decremented if they disagree.

References

[1] D.A. Jimenez and C. Lin. Dynamic branch prediction with perceptrons. In *Proceedings HPCA Seventh International Symposium on High-Performance Computer Architecture*, pages 197–206, 2001.