

Universidad de Costa Rica

Facultad de Ingeniería

Escuela de Ingeniería Eléctrica

IE0521 – Estructuras de Computadoras II

I ciclo 2024

Tarea 1

Lorena Solís Extenny B97657

Ronny Granados Perez C03505

Profesor: Erick Carvajal Barboza

3 de Mayo, 2024

Predictor de Saltos P-Shared

Resultados P-shared

Primeramente, se generaron los resultados del programa para el predictor de saltos P-shared de manera que se variaron tanto la cantidad de bits del PC a indexar como el tamaño de la historia local y se muestran los mismos en la siguiente tabla:

Tamaño historia local	Bits del PC para indexar				
	4	8	12	16	20
2	66.84 %	74.22 %	84.93 %	86.59 %	86.69 %
6	68.54 %	76.42 %	88.02 %	89.48 %	89.58 %
8	69.71 %	77.38 %	89.10 %	90.51 %	90.59 %
16	84.61 %	85.82 %	92.49 %	93.50 %	93.55 %
20	87.91 %	86.46 %	92.39 %	93.40 %	93.45 %

Tabla 1: Experimentos predictor p-shared

Posteriormente, a partir de estos datos se genera el gráfico 1 que muestra como se comportan el porcentaje de predicciones correctas según la cantidad de bits a indexar y la variación del tamaño de la historia local:

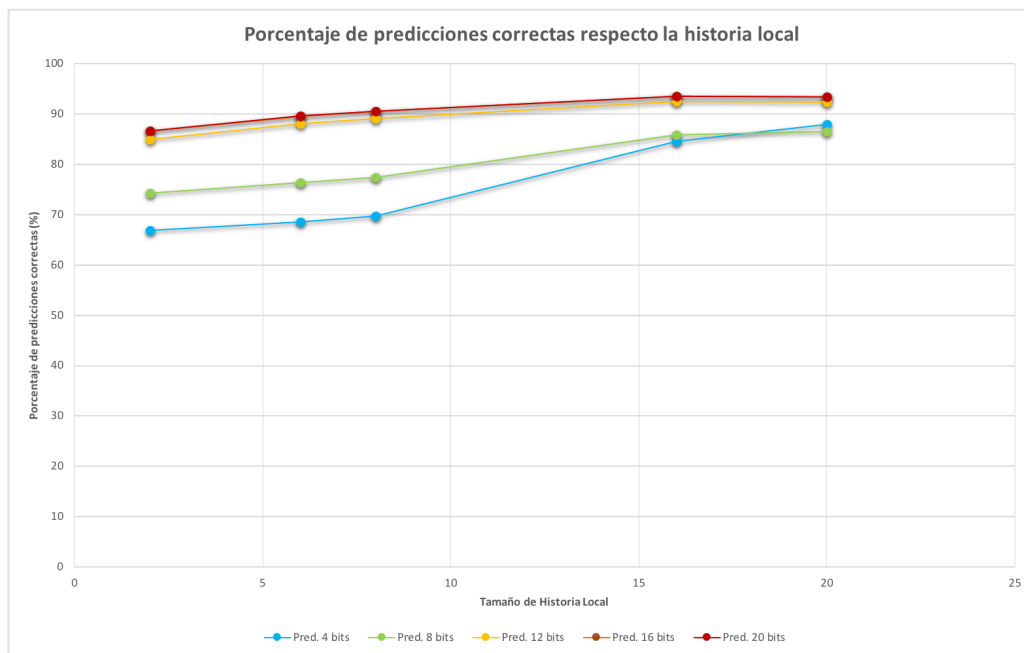


Figura 1: Porcentaje de predicciones correctas respecto a la historia local.

Análisis de datos

-Si la cantidad de bits del PC se mantiene constante, ¿cómo cambia la precisión ante cambios en el tamaño del registro de historia?

Se observa que al aumentar el tamaño del registro de historia, el predictor experimenta una mejora en su precisión. Por ejemplo, para un tamaño de registro de historia de 4 bits del PC, la precisión aumenta hasta un 21,07 % al cambiar de un tamaño de historia local de 2 a 20. Sin embargo, si los bits del PC son 20, el cambio en la precisión al variar el tamaño de la historia local de 2 a 16 es solo del 6,86 %. Esta mejora se debe a que un registro de historia más grande

permite al predictor recordar patrones más complejos de comportamiento de los saltos. Aunque la mejora es notable, es importante considerar que existe un punto de saturación donde añadir más información histórica puede tener un retorno decreciente en la precisión y aumentar la complejidad computacional del predictor.

-Si la cantidad de bits del registro de historia local se mantiene constante, ¿cómo cambia la precisión ante cambios en la cantidad de bits del PC utilizados para indexar?

En este caso, se observa que, al aumentar la cantidad de bits del PC utilizados para indexar, la precisión del predictor tiende a mejorar. Por ejemplo, con un tamaño de 2 de la historia local, hay una diferencia de hasta un 19,85 % en la precisión al aumentar los bits del PC de 4 a 20. Sin embargo, con un tamaño de historia local de 20, la variación entre 4 y 20 bits es solo del 5,54 %. Este aumento en la precisión se debe a que una mayor cantidad de bits del PC permite una indexación más precisa, lo que facilita la identificación de patrones en el comportamiento de los saltos. Al igual que en el caso anterior, es importante tener en cuenta que existe un punto de saturación donde añadir más bits del PC puede tener retornos decrecientes en la precisión y aumentando la complejidad del predictor.

Predictor de saltos basado en Perceptrones

Resultados Perceptrones

Ahora, se muestran los resultados obtenidos al experimentar con un predictor de saltos basado en perceptrones:

Bits de historia global	Bits del PC para indexar				
	4	8	12	16	20
2	71,49	76,85	81,24	84,38	86,17
6	74,26	81,6	87,05	90,02	90,98
8	74,78	83,01	88,53	91,28	92,22
16	76,45	86,12	91,48	93,69	94,47
20	77,01	86,99	92,2	94,15	94,89

Tabla 2: Experimentos predictor basado en perceptrones

Seguidamente, se muestra el gráfico 2 que muestra el comportamiento del predictor con aumentos de bits del PC y de la historia global:

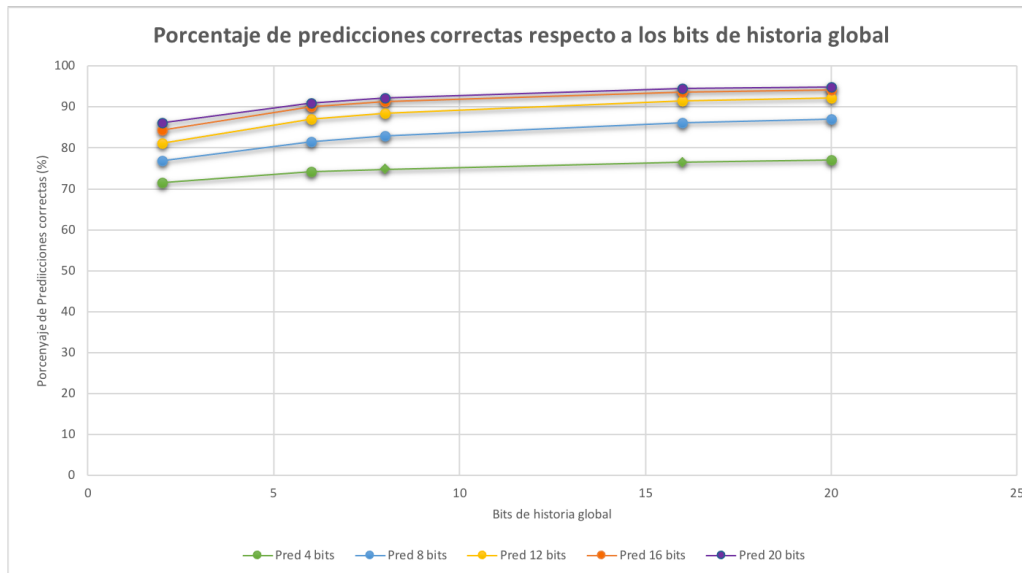


Figura 2: Porcentaje de predicciones correctas respecto a la historia global.

Análisis de datos

-Si la cantidad de bits del PC se mantiene constante, ¿cómo cambia la precisión ante cambios en el tamaño del registro de historia?

Al igual que con el predictor de saltos pshared, se observa una mejora en la precisión del predictor basado en perceptrones al aumentar el tamaño del registro de historia global. Por ejemplo, al mantener constante la cantidad de bits del PC en 20, la precisión aumenta hasta un 8,72 % al pasar de una cantidad de bits de historia global de 2 a 20. En este caso las variaciones cambian dependiendo del valor de PC que se mantenga constante, ya que, por ejemplo en el caso en que los bits del PC son 16 se nota el mayor cambio a la hora de modificar los bits de la historia global con una mejora del 10,96 %.

-Si la cantidad de bits del registro de historia global se mantiene constante, ¿cómo cambia la precisión ante cambios en la cantidad de bits del PC utilizados para indexar?

Al mantener constante la cantidad de bits del registro de historia global, se observa que la precisión del predictor también aumenta al incrementar la cantidad de bits del PC utilizados para indexar. Por ejemplo, con un tamaño de historia global de 2 bits, la precisión varía hasta un 14,68 % al aumentar los bits del PC de 4 a 20. Sin embargo, al incrementar el tamaño de la historia global, esta mejora tiende a ser menos significativa, como se puede observar desde los bits del PC del 12-20 y se podría decir que del 4-12 se muestran los cambios más significativos.

Predictor de saltos propuesto

Propuesta

Funcionamiento

El predictor propuesto consiste en dos partes: una tabla de registros desplazantes de historia local que funcionan como índices a una tabla de perceptrones con los cuales, una vez entrenados, se puede predecir el comportamiento que tomarán los branches. Explicado a mayor detalle, el

predicador toma como parámetros la cantidad de bits que se desean indexar (sea esto N) y el tamaño de los registros de historia local (llámesele M). A partir de esta información se inicializa una lista a la que se llamará **Tabla PC**, debido a que la forma de acceder a esta es mediante el contador de programa (PC), de tamaño 1×2^N . Por otra parte, con la información del tamaño de la historia local se inicializa la tabla que almacenará los perceptrones, llámesele **Tabla de Perceptrones**, esta tabla es de tamaño $2^M \times M$.

Una vez que se inicia la lectura del archivo **.gz** con la información de los saltos se extrae de este el PC y se modula con la cantidad de entradas de la Tabla PC para poder así acceder a dicha tabla. Dentro de la Tabla PC se encontraran en cada una de las entradas un registro desplazante de M bits con el cual se obtiene el índice para acceder al M -ésimo perceptrón y, con la información almacenada de sus pesos, predecir qué es lo que se hará con el branch: se tomará o no se tomará.

Justificación de funcionalidad

Implementar una tabla de perceptrones no es una tarea complicada y se sabe de antemano que funcionan bastante bien para predecir los saltos, por lo tanto, para la implementación de este predictor se decidió apoyarse de este tipo de red neuronal en conjunto con una tabla de registros de historia local en lugar de acceder a los perceptrones directamente. Su funcionamiento es exitoso puesto que se mantiene dentro del presupuesto de hardware (discutido y desglosado más adelante) y muestra un buen desempeño al ofrecer un alto porcentaje de predicción y un tiempo de ejecución corto (de 2.5 minutos aproximadamente).

Cálculo de presupuesto

Para el cálculo del presupuesto se toman en cuenta todas aquellas estructuras que demanden bits en su implementación. En el caso de este predictor de saltos, se toman en cuenta los bits de cada uno de los registros desplazantes de historia local en la Tabla PC y cada uno de los pesos de los perceptrones en la Tabla de Perceptrones, siendo que para cada peso se asignan 7 bits con signo. Se tiene entonces que el hardware total, en bits, se calcula de la siguiente manera:

$$PRESUPUESTO = Tabla PC + Tabla Perceptrones$$

$$PRESUPUESTO = M \cdot 2^N + 7 \cdot 2^M \cdot (M + 1)$$

El presupuesto establecido en el enunciado para este predictor es de 32 kb, si se define un tamaño fijo de bits para indexar y de historia local de $N = 11$ bits y $M = 8$ bits respectivamente, según la ecuación descrita previamente, se obtiene lo siguiente:

$$PRESUPUESTO = 8 \cdot 2^{11} + 8 \cdot 2^8 \cdot (8 + 1) = 32512 \text{ bits}$$

Lo cual permanece dentro del rango establecido (presupuesto: 32768 bits) y además, da un muy buen porcentaje de predicción, siendo este de **88.39 %**.

Parámetros del predictor:	
Tipo de predictor:	Perceptron-Shared
Entradas en la Tabla PC:	2048
Tamaño de los registros de historia global:	8
Resultados de la simulación	
# branches:	16416279
# branches tomados predichos correctamente:	5206088
# branches tomados predichos incorrectamente:	1005507
# branches no tomados predichos correctamente:	9304895
# branches no tomados predichos incorrectamente:	899789
% predicciones correctas:	88.394%

Figura 3: Resultados Obtenidos para el Predictor Propuesto.

Para conocer más de la construcción de este predictor se proporciona el código fuente, así como el archivo para probarlo **.gz**.