

Universidad de Costa Rica
Facultad de Ingeniería
Escuela de Ingeniería Eléctrica
IE 0311 – Estructuras de Computadores Digitales 2
I ciclo 2024

Tarea 1: Predicción de Saltos

Melissa Rodríguez Jiménez C16634
Leonardo Leiva Vasquez C14172

Profesor:
Erick Carvajal Barboza

3 de mayo del 2024

En el presente reporte se incluye el análisis de resultados obtenidos de las distintas simulaciones para los predictores de saltos P-Shared y por Perceptrones. También se incluye un tercer predictor de saltos propuesto por los estudiantes, al igual que una descripción de su funcionamiento y su precisión. Los códigos de cada algoritmo se encuentran en los archivos *pshared.py*, *perceptron.py* y *ie0521_bp.py*, respectivamente. En el archivo de README se encuentra una breve descripción para correr estos programas y revisar las simulaciones.

Predictor P-Shared

El predictor de saltos P-Shared consiste en grandes rasgos en dos tablas, una de historia local y una de patrones, que le permite decidir si se toma o no el salto. Cierta cantidad de bits del PC permite indexar la primera tabla con registros de historia local deslizantes, los cuales a su vez indexan la segunda tabla de patrones que contiene propiamente el control de la historia local y modifica los valores cada vez que se toma o no un salto (modificándose también los registros deslizantes).

Luego de la codificación de este algoritmo, se realizaron distintas pruebas variando la cantidad de bits del PC y de la historia local para analizar el rendimiento de la predicción. La tabla 1 muestra los porcentajes de predicciones correctas para cada combinación, y a su vez, la Figura 2 muestra de forma gráfica dichos resultados.

Tamaño historia local	Bits del PC para indexar				
	4	8	12	16	20
2	66.84 %	74.22 %	84.94 %	86.60 %	86.69 %
6	68.54 %	76.43 %	88.03 %	89.49 %	89.58 %
8	69.71 %	77.39 %	89.11 %	90.52 %	90.59 %
16	84.62 %	85.82 %	92.50 %	93.51 %	93.56 %
20	87.91 %	86.46 %	92.39 %	93.40 %	93.45 %

Tabla 1: Porcentaje de predicciones correctas para distintos bits de PC y de historia local

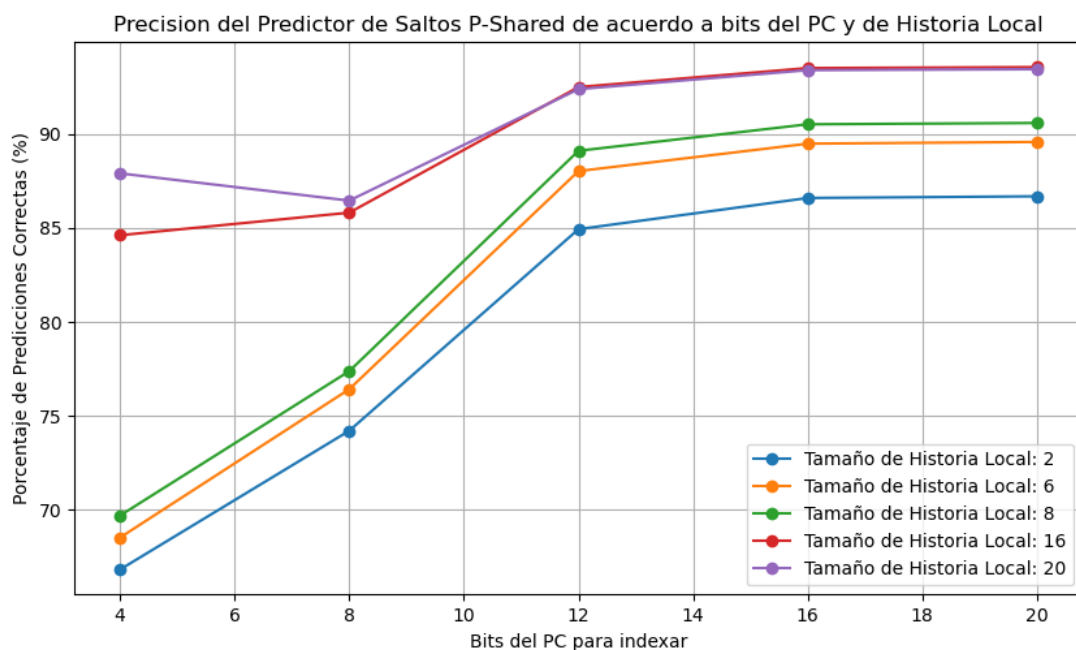


Figura 1: Gráfica de porcentaje de predicciones correctas del P-Shared

Análisis de Tendencias

Manteniendo la Cantidad de Bits del PC Constante

Cuando se mantiene constante los bits del PC y se va cambiando los bits de la historia local, se observa como la precisión va aumentando de poco a poco al inicio, luego aumenta abruptamente y luego vuelve con pequeños aumentos. Por ejemplo, con 4 bits de PC, la precisión no llega a aumentar ni el 2% respecto al valor anterior, esto cuando va cambiando entre los 2 y los 8 bits. Sin embargo, al pasar de 8 bits de historia local a 16 se observa un mayor cambio en la precisión. Para los otros casos se aprecia un comportamiento similar pero a menor escala. Lo anterior se puede apreciar mejor en la gráfica realizada, nótese que las líneas celeste, naranja y verde están bastante juntas (representan el tamaño de la historia con 2, 6 y 8 bits respectivamente) mientras que la roja y la morada están juntas entre si, pero separadas de las demás (tamaño de 16 y 20 bits).

Manteniendo la Cantidad de Bits del Registro de Historia Local Constante

Cuando se mantiene constante el tamaño de la historia local y se varía la cantidad de bits del PC se observa una tendencia a aumentar la precisión rápidamente al inicio y luego mantenerse constante. Lo anterior se puede visualizar en la gráfica, observando que las líneas en la mayoría de los casos van creciendo casi de 10 en 10, hasta llegar a los 12 bits del PC donde el porcentaje comienza a regularse y aumentar poco.

Predictor Basado en Perceptrones

Segun el artículo «Dynamic Branch Prediction with Perceptrons» el algoritmo del perceptrón es una técnica utilizada para predecir decisiones en un programa. Funciona a través de una red neuronal artificial simple llamada perceptrón, que aprende de ejemplos pasados para predecir decisiones futuras. Los pesos internos del perceptrón se ajustan durante el entrenamiento para aprender patrones en los datos y luego se utilizan para hacer predicciones sobre nuevas situaciones. La entrada para el perceptrón en este contexto incluye características como los bits del Program Counter (PC) y el registro de historia global de ramas, y el perceptrón utiliza estos datos para predecir si un salto será tomado o no.

En la siguiente tabla se muestra el porcentaje de predicciones correctas para diferentes configuraciones de bits del PC y tamaño de historia local. Se puede observar cómo varía la precisión de las predicciones al ajustar estos parámetros en el algoritmo de predicción de saltos por medio de perceptrón.

Tamaño historia Global	Bits del PC para indexar				
	4	8	12	16	20
2	71.47 %	84.06 %	90.67 %	91.35 %	91.36 %
6	74.46 %	89.56 %	93.38 %	93.64 %	93.66 %
8	75.16 %	90.69 %	93.99 %	94.22 %	94.24 %
16	77.01 %	92.92 %	95.51 %	95.71 %	95.72 %
20	77.56 %	93.35 %	95.79 %	95.99 %	96.00 %

Tabla 2: Porcentaje de predicciones correctas para distintos bits de PC y de historia global para el Perceptron

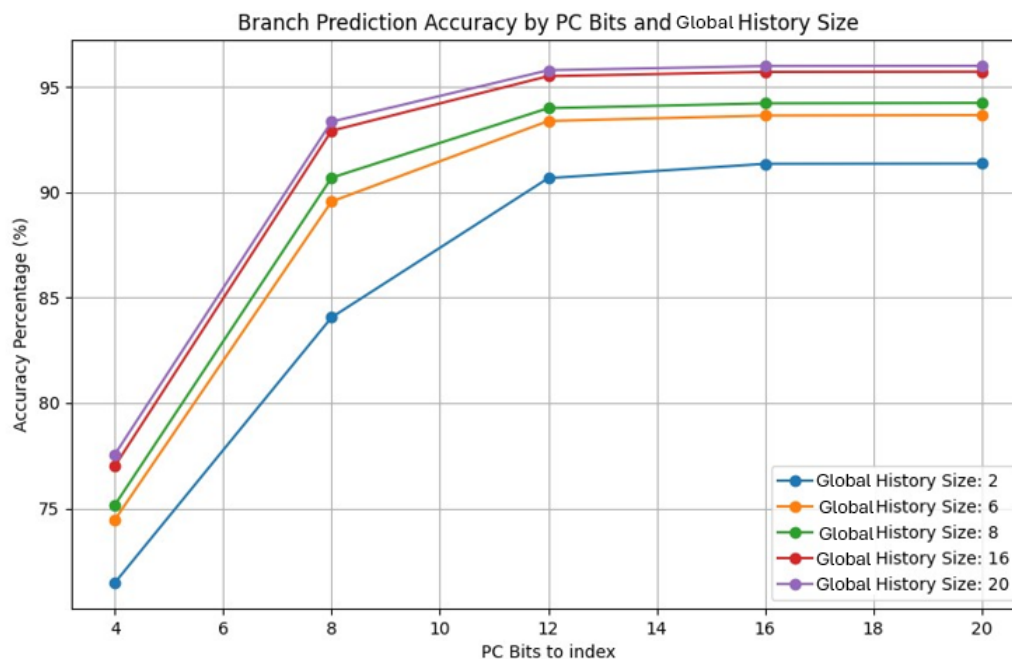


Figura 2: Gráfica de porcentaje de predicciones correctas del algoritmo basado en perceptrón

Análisis de Tendencias

Manteniendo la Cantidad de Bits del PC Constante

- A medida que aumenta el tamaño del registro de historia global, generalmente se observa un aumento en la precisión de las predicciones.
- Esto se debe a que un registro de historia global más grande proporciona más información histórica sobre las decisiones de ramas anteriores, lo que ayuda al perceptrón a aprender mejor los patrones y a realizar predicciones más precisas.

Manteniendo la Cantidad de Bits del Registro de Historia Global Constante

- En general, se observa que aumentar la cantidad de bits del PC utilizados para indexar conduce a mejoras en la precisión de las predicciones.
- Esto se debe a que una mayor cantidad de bits del PC permite una indexación más precisa de la información histórica en el registro de historia global.
- Con más bits del PC, el perceptrón puede distinguir mejor entre diferentes patrones de comportamiento de ramas y ajustar sus predicciones de manera más precisa.

Algoritmo de Predicción de Saltos Propuesto

Explicación de Funcionamiento

El predictor propuesto combina dos técnicas de predicción de saltos: el G-Share para la variación de la historia global y un algoritmo de perceptrones utilizado en la tarea. El primer componente clave es el método G-Share. Este enfoque implica variar el valor de la historia global, que representa los patrones de comportamiento de los saltos en el programa. Utiliza una operación XOR entre la historia global y el contador de programa (PC) para generar un índice.

Este índice se utiliza para acceder a una matriz de pesos que contiene información sobre las decisiones de salto anteriores y su efectividad.

El segundo componente es el algoritmo de perceptrones, que se basa en el concepto de una red neuronal simplificada. En este contexto, el algoritmo de perceptrones se utiliza para actualizar los pesos de la matriz en función de las entradas proporcionadas por el historial global y el estado actual del PC.

En cada actualización, tanto el historial global del G-Share como el índice del contador de programa (PC) toman valores diferentes y esto propone una ventaja en su capacidad para introducir una mayor variabilidad en la elección de pesos. Al utilizar el XOR entre el índice del PC actualizado y la historia global actualizada, el predictor puede capturar patrones más complejos y sutiles en los datos de historial de saltos.

Justificación de funcionalidad

En cada iteración de actualización del predictor, tanto el historial global del G-Share como el índice del contador de programa (PC) actualizan sus valores. Esta dinámica es importante ya que proporciona una ventaja significativa al predictor en términos de su capacidad para introducir una variabilidad más amplia en la elección de pesos. La variabilidad en la elección de pesos es muy importante porque permite que el predictor se ajuste mejor a los cambios y a los patrones complicados que aparecen en los datos de historial de saltos. Lo cual significa que puede adaptarse más fácilmente a cómo el programa se comporta en diferentes situaciones.

Esto se logra gracias a la introducción del operador XOR entre el índice del PC actualizado y la historia global actualizada. Este operador XOR, al comparar y combinar los valores de manera exclusiva entre el índice del PC y la historia global, ayuda al predictor a notar y entender patrones más complicados y detallados en la forma en que el programa toma decisiones sobre los saltos en su ejecución.

La mejora en la predicción de saltos se logra gracias a la combinación de una mayor variabilidad en el índice de la matriz y la configuración de los pesos en el predictor propuesto en comparación con otros predictores de salto. Por ejemplo, al utilizar un índice de 20 bits y un tamaño de historial global de 20 (caso que se sale del presupuesto, pero se menciona para efectos comparativos), se observa un rendimiento superior en el predictor propuesto en comparación con el perceptron y el g-share.

Los resultados muestran que el perceptron logra un 96 % de aciertos, mientras que el g-share obtiene un 93.45 %. En contraste, el predictor propuesto alcanza un 96.36 % de aciertos. Esto significa que el predictor propuesto tiene una precisión aproximadamente 61200 saltos mayor que el perceptron en la predicción de saltos.

Cálculo de presupuesto

Para el cálculo del presupuesto se tomó en consideración el uso de la matriz de pesos, cuya cantidad de filas corresponde a $2^{Bits-PC}$ y de columnas corresponde a la cantidad de bits de la historia global más 1 (para guardar el bios o peso inicial). Es por eso que el presupuesto de este predictor propuesto se calcularía de la siguiente manera:

$$\text{Presupuesto} = 2^{Bits-PC} \times (\text{Bits Historia Global} + 1)$$

Para no excederse del límite de presupuesto (32768 bits) se optó por mantener el número de bits del PC en 12 y el de la historia global en 6, de modo que el presupuesto diera 28672 bits, cumpliendo así con lo solicitado. La Figura 3 muestra los resultados obtenidos con dicha cantidad de bits para el PC y para la historia global. Nótese que la precisión es bastante alta, con un 93.60%, considerando que no se utilizó una gran cantidad de bits. Por lo que de esta forma se comprueba su funcionamiento y la efectividad de unir técnicas de varios predictores ya existentes.

```
python3 branch_predictor.py --bp 4
Parámetros del predictor:
    Tipo de predictor:          IE0521_BP
    Entradas en el Predictor:   4096
    Tamaño de los registros de historia global: 6
Resultados de la simulación
    # branches:                  16416279
    # branches tomados predichos correctamente: 5634092
    # branches tomados predichos incorrectamente: 577503
    # branches no tomados predichos correctamente: 9730849
    # branches no tomados predichos incorrectamente: 473835
    % predicciones correctas:    93.60%
```

Figura 3: Resultados de la simulación con Predictor Propuesto