

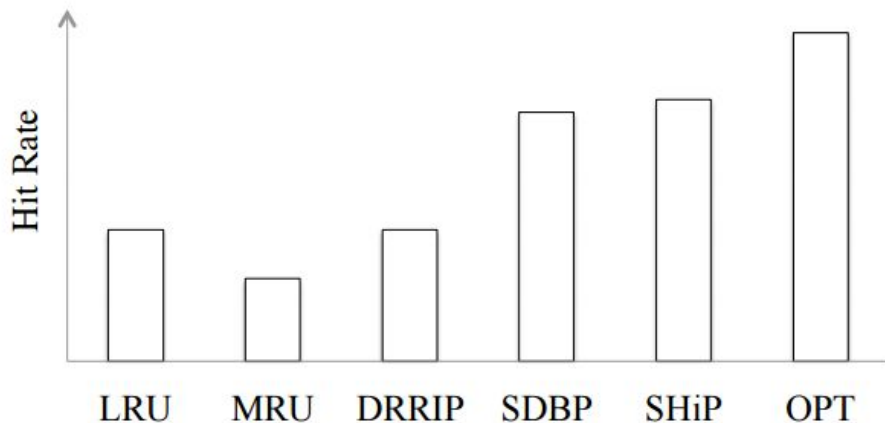
Back to the Future:
Leveraging Belady's
Algorithm for Improved
Cache Replacement
- ISCA 2016 -

Introdução

Eficiência da cache é significativamente influenciada pela política de substituição (replacement policy).

As políticas existentes baseiam-se em heurísticas, como MRU e LRU.

- Bom desempenho para diferentes cargas de trabalho.



Algoritmo OPT

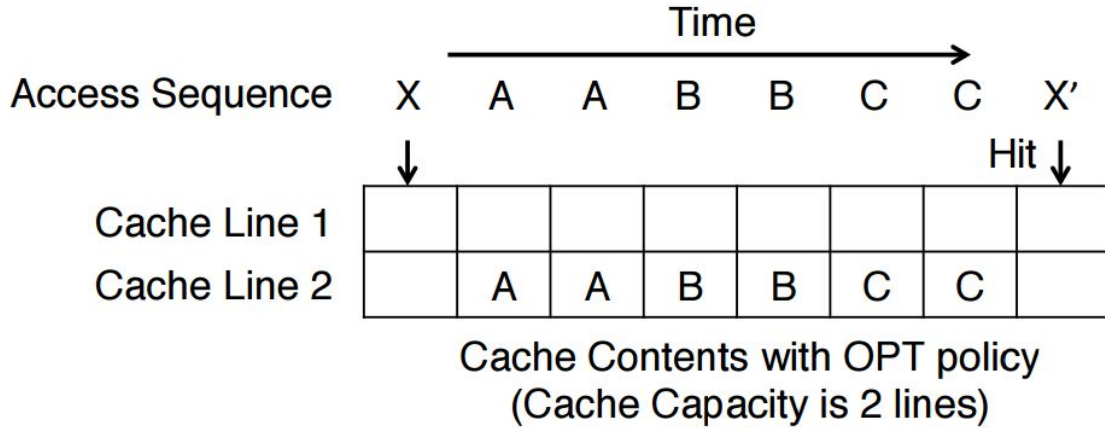


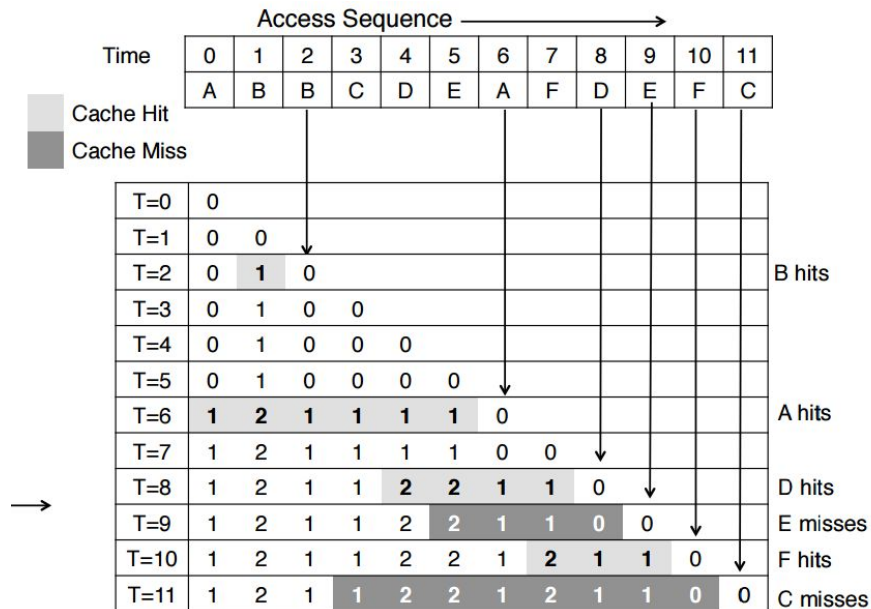
Figure 5: Intuition behind OPTgen.

Key Ideas

- Não podemos olhar para o futuro.
- Aplicar OPT aos eventos passados para aprender como o OPT teria se comportado.
- Se passado prediz o futuro, então essa solução deve se aproximar de OPT.

Example OPTgen

- Calcula se uma linha eh cache-friendly or averse baseado no algoritmo OPT.
(Se o OPT estivesse sendo utilizado, essa linha seria cacheada ou não?)



(c) OPTgen Solution (4hits)
[State of the Occupancy Vector over time]

Preditor

Aprende quando uma dada instrução de load resultaria em HIT ou MISS baseado no OPTgen.

- Se OPTgen diz que haverá hit -> treinado positivamente.
- Se OPTgen diz que haverá miss -> treinado negativamente.

RRIP alto indica candidatos a serem retirados da cache. Há feedback para saber se ouve hit ou miss para as predições feitas.

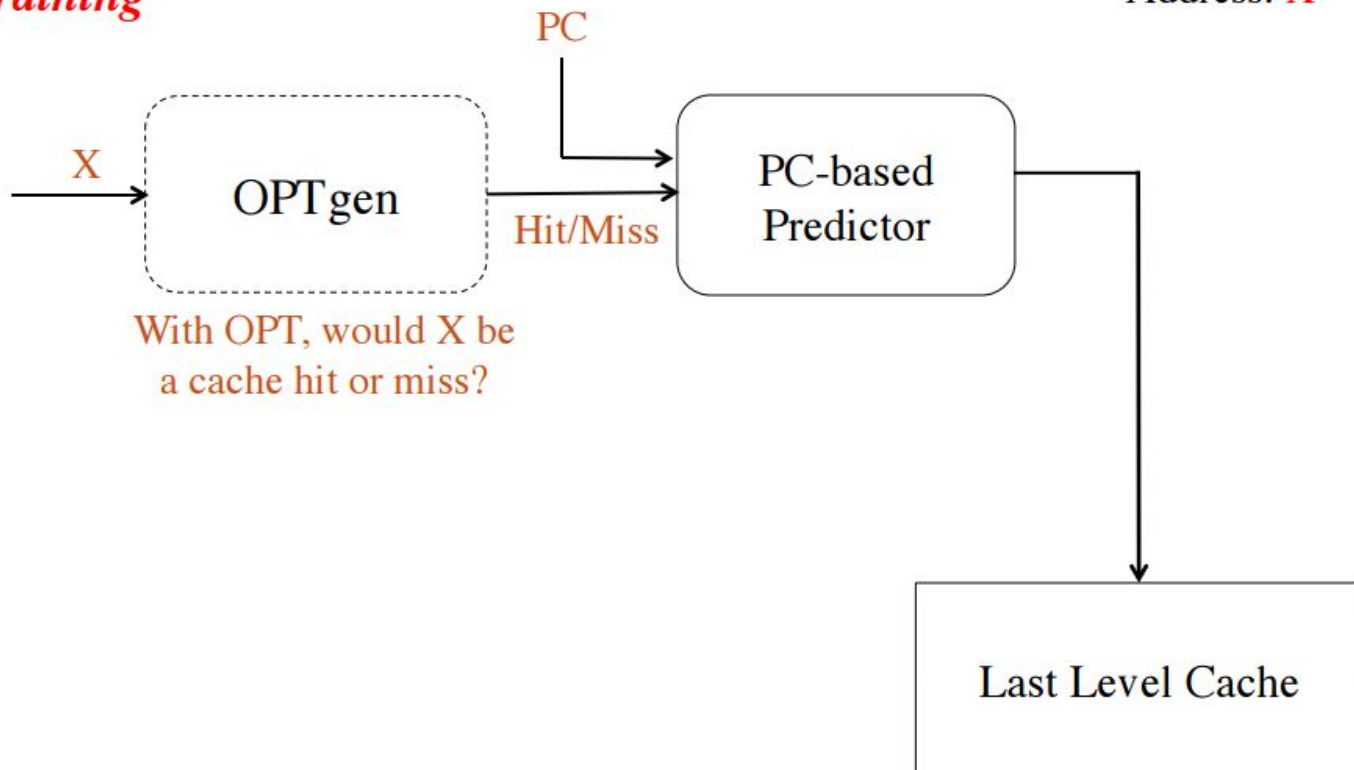
Hawkeye Prediction \ Hit or Miss	Cache Hit	Cache Miss
	RRIP = 7	RRIP = 7
Cache-averse	RRIP = 7	RRIP = 7
Cache-friendly	RRIP = 0	RRIP = 0; Age all lines: if (RRIP < 6) RRIP++;

Table 1: Hawkeye's Update Policy.

Hawkeye: Overall Design

Training

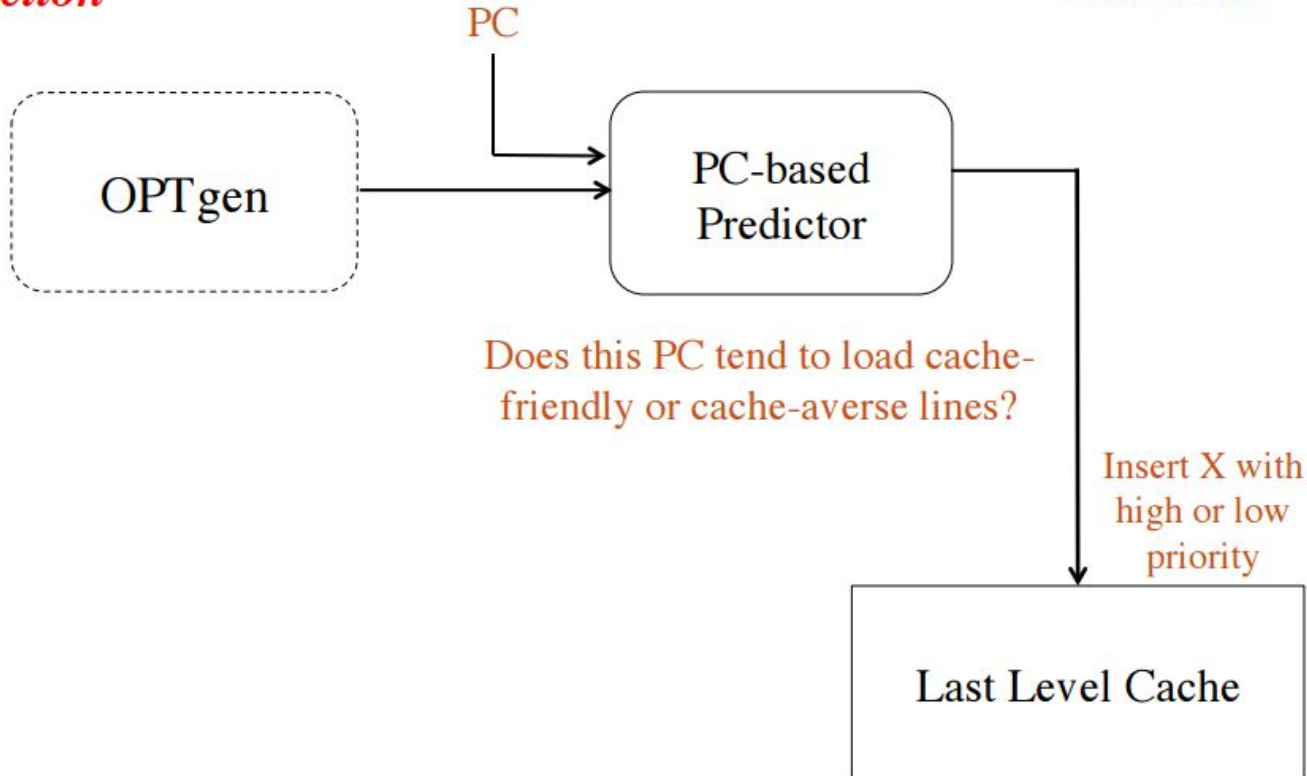
Address: **X**



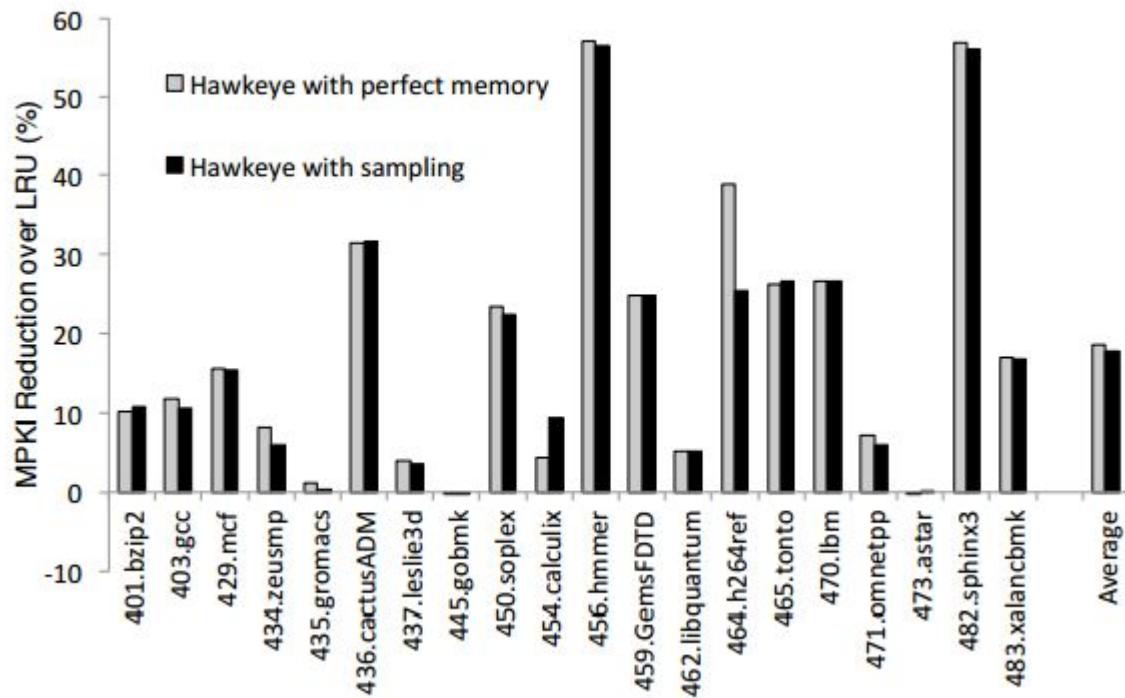
Hawkeye: Overall Design

Prediction

Address: **X**



O que?



Como?

First JILP Cache Replacement Championship (CRC)

Referências

<https://www.cs.utexas.edu/~lin/papers/isca16.pdf>

<http://isca2016.eecs.umich.edu/wp-content/uploads/2016/07/2A-1.pdf>