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

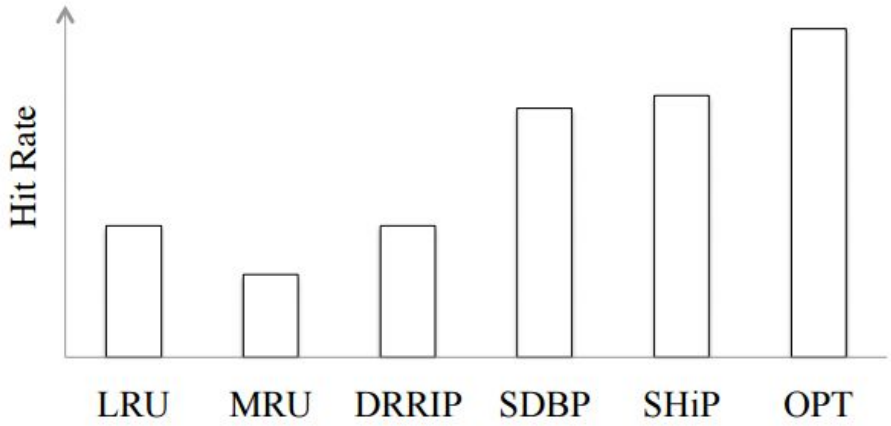
Alceu Emanuel Bissoto - 191077

Introdução

Eficiência da cache é significantemente 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

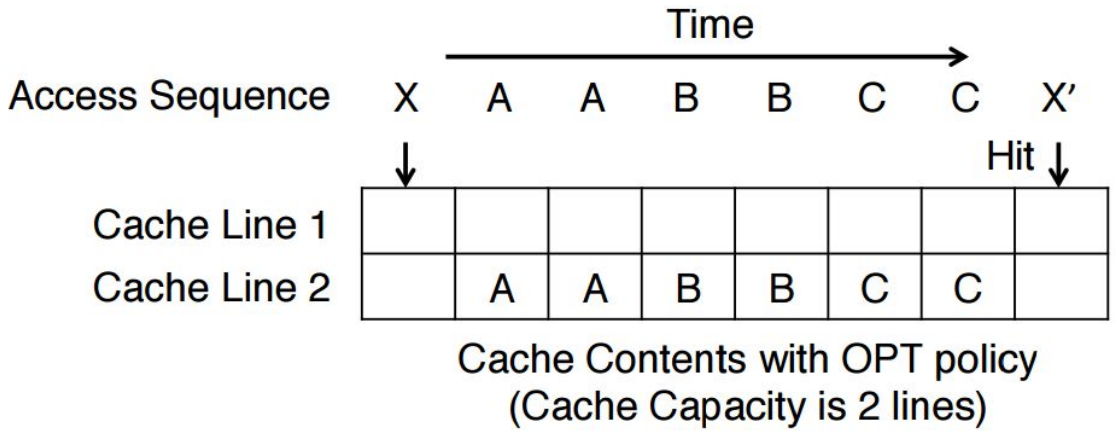
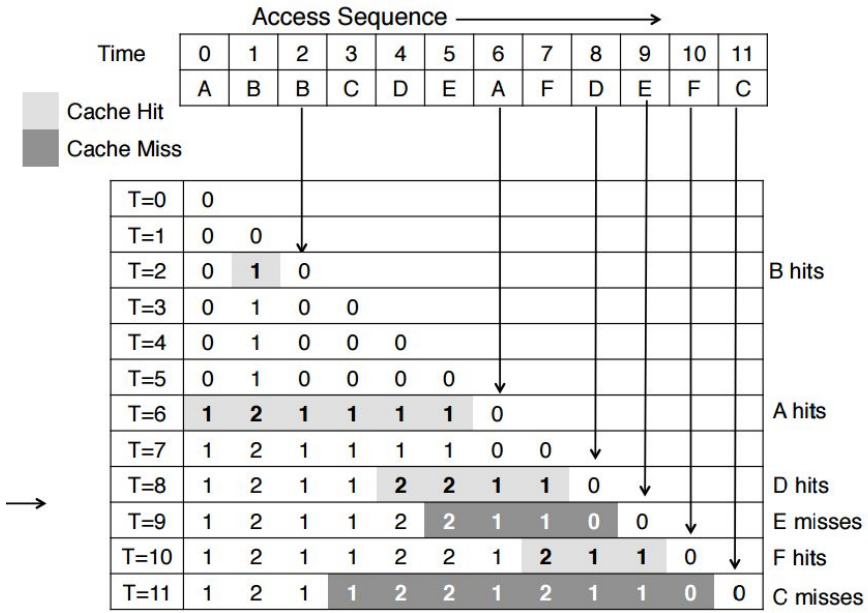


Figure 5: Intuition behind OPTgen.

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]

Sniper

getReplacementIndex()

-Utilizado para retornar qual way deve ser retirada da cache.

-LRU

-Chamada quando há possibilidade

da necessidade de substituição de

dados da cache.

updateReplacementIndex()

-Utilizada para atualização das estruturas de controle do algoritmo.

-OPTgen

-Chamada a toda leitura/escrita na cache.

Implementação

Vetores:

M_occupancy : vetor de ocupação; 128 posições.

M_last_occurrence : armazena a última posição em que apareceu no vetor de ocupação para cada way desta linha de cache.

M_priority_eviction : score para cada way desta linha de cache, onde mais alto significa maior prioridade de ser retirada.

Sniper

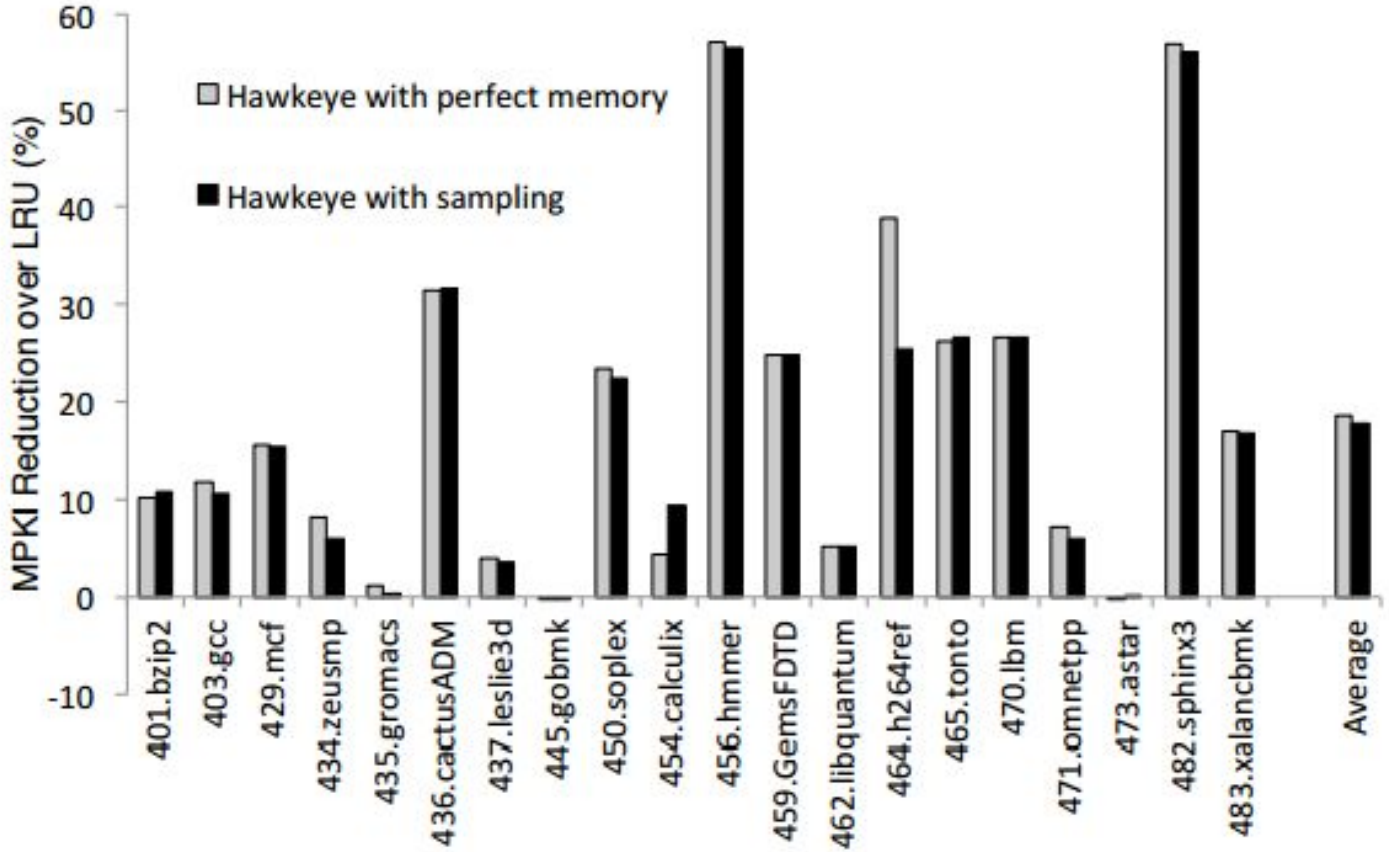
getReplacementIndex()

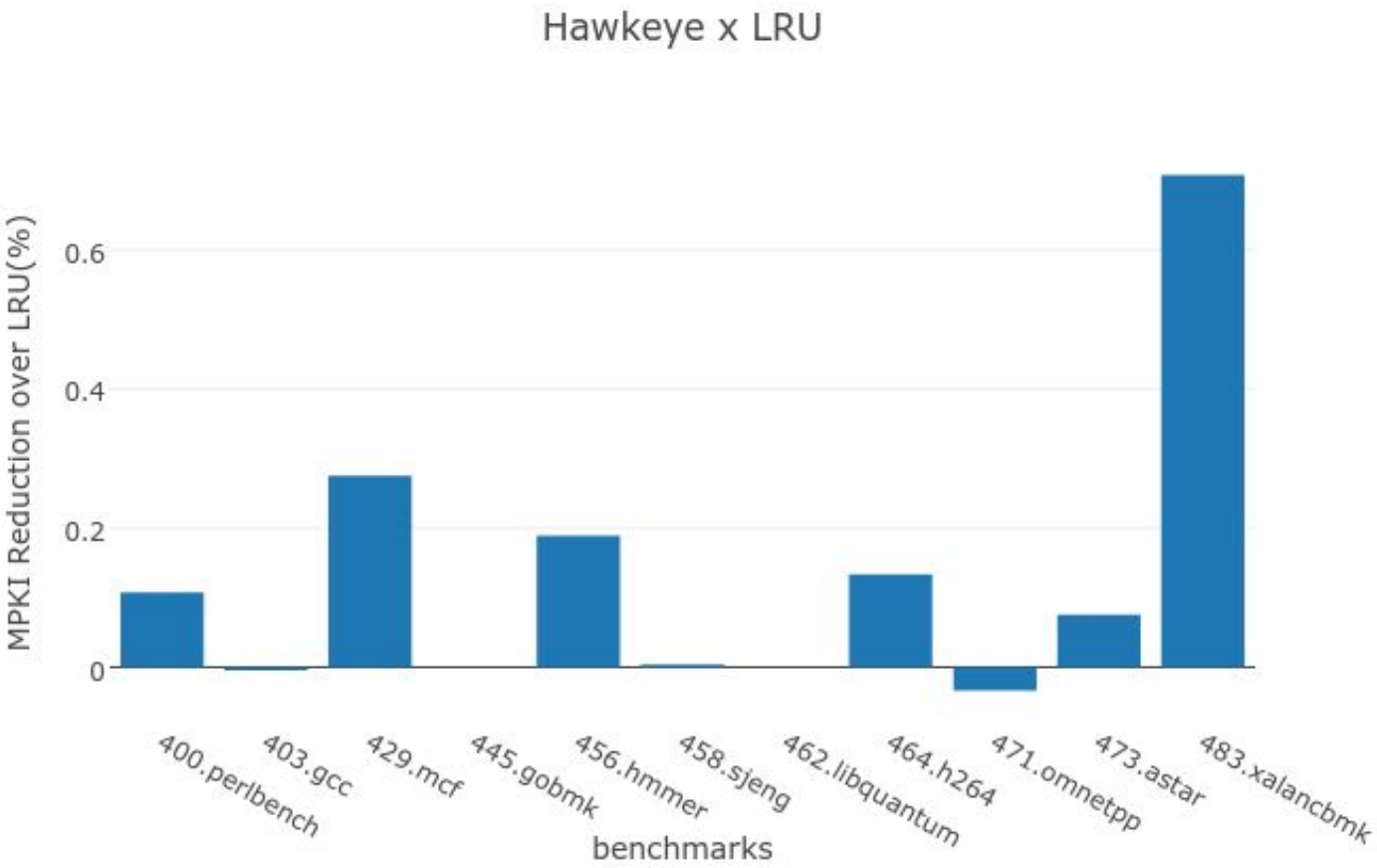
- Consulta o m_priority_eviction, e retorna o indice da via com o maior score.
- Modifica o vetor de modo a trabalhar como LRU, caso não haja predição de miss por parte do OPTgen.

updateReplacementIndex()

Verifica a cada escrita, se teria havido miss ou hit sobre a politica OPT.

Modifica o m_priority_eviction, dando prioridade maxia para retirada para predições de misses.


MPKI Reduction over LRU (%)
60
50
40
30
20
10
0
-10
Hawkeye with perfect memory
Hawkeye with sampling
401.bzip2
403.gcc
429.mcf
434.zeusmp
435.gromacs
436.cactusADM
437.leslie3d
445.gobmk
450.soplex
454.calculix
456.hmmer
459.GemsFDTD
462.libquantum
464.h264ref
465.tonto
470.lbm
471.omnetpp
473.astar
482.sphinx3
483.xalancbmk
Average


Hawkeye x LRU
MPKI Reduction over LRU(%)
0.6
0.4
0.2
0
400.perlbench
403.gcc
429.mcf
445.gobmk
456.hmmer
458.sjeng
462.libquantum
464.h264
471.omnetpp
473.astar
483.xalancbmk
benchmarks

Referências

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

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