Cache Related Preemption Delay Computation for Set-Associative Caches

Pitfalls and Solutions

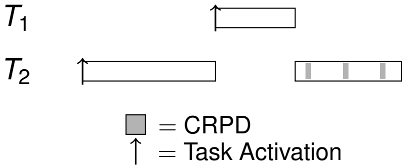
Claire Burguière, Jan Reineke, Sebastian Altmeyer

Workshop on WCET Analysis, Dublin 2009

SAARLAND UNIVERSITY
COMPUTER SCIENCE

Context

- Preemptive scheduling
- Cache related preemption delay (CRPD):
  - Impact of preemption on the cache content
  - Overall cost of additional reloads due to preemption



CRPD computation for set-associative caches

- CRPD computation:
  - Useful Cache Blocks (UCB)
  - Evicting Cache Blocks (ECB)
- CRPD for set-associative caches:
  ⇒ Pitfalls:
  - LRU: CRPD not bounded by the number of ECBs
  - FIFO and PLRU: CRPD not bounded

Useful Cache Block - [Lee et al., 1996]

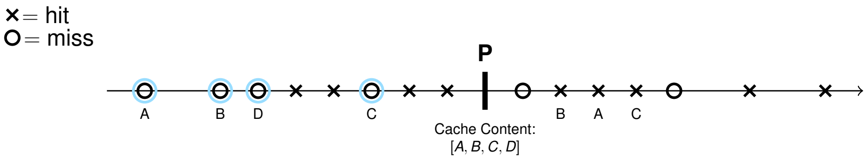
SAARLAND UNIVERSITY
COMPUTER SCIENCE

Definition (Useful Cache Block)

A memory block m at program point P is called a useful cache block, if

a) *m may be cached at P*

b) *m may be reused at program point P′ that may be reached from P with no eviction of m on this path.*



Useful Cache Block - [Lee et al., 1996]

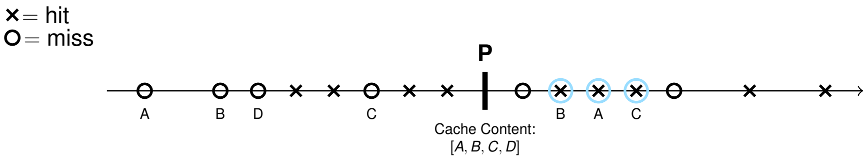
SAARLAND UNIVERSITY
COMPUTER SCIENCE

Definition (Useful Cache Block)

A memory block m at program point P is called a useful cache block, if

a) *m may be cached at P*

b) *m may be reused at program point P′ that may be reached from P with no eviction of m on this path.*



Useful Cache Block - [Lee et al., 1996]

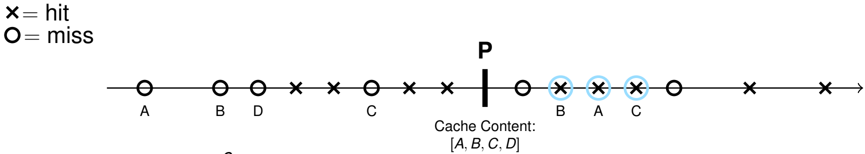
SAARLAND UNIVERSITY
COMPUTER SCIENCE

Definition (Useful Cache Block)

A memory block m at program point P is called a useful cache block, if

a) *m may be cached at P*

b) *m may be reused at program point P' that may be reached from P with no eviction of m on this path.*



$$\text{CRPD}^{\text{LRU}} = \sum_{s=1}^{c} \text{CRPD}^{\text{LRU}}(s)$$

$$\text{CRPD}^{\text{LRU}}_{\text{UCB}}(s) = \text{BRT} \cdot min(|\text{UCB}(s)|, n)$$

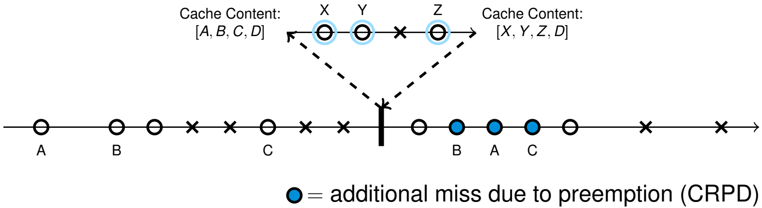
n = associativity
BRT = Block Reload Time

Evicting Cache Blocks [Tomiyama and Dutt, 2000]

SAARLAND UNIVERSITY
COMPUTER SCIENCE

Definition (Evicting Cache Blocks (ECB))

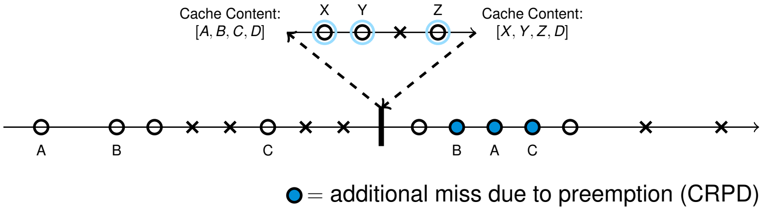
A memory block of the preempting task is called an evicting cache block, if it may be accessed during the execution of the preempting task.



Evicting Cache Blocks [Tomiyama and Dutt, 2000]

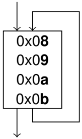
Definition (Evicting Cache Blocks (ECB))

A memory block of the preempting task is called an evicting cache block, if it may be accessed during the execution of the preempting task.



$$\mathrm{CRPD}^{\mathrm{LRU}}_{\mathrm{ECB}}(s) \stackrel{?}{=} \mathrm{BRT} \cdot min(|\mathrm{ECB}(s)|, n)$$

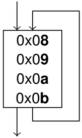
CRPD computation for LRU using ECB: Pitfall



$[b, a, 9, 8] \xrightarrow{8} [8, b, a, 9] \xrightarrow{9} [9, 8, b, a] \xrightarrow{a} [a, 9, 8, b] \xrightarrow{b} [b, a, 9, 8]$ 0 misses

CRPD computation for LRU using ECB: Pitfall

SAARLAND UNIVERSITY
COMPUTER SCIENCE



ECBs $= \{\mathbf{e}\}$

$$[b, a, 9, 8] \xrightarrow{8} [8, b, a, 9] \xrightarrow{9} [9, 8, b, a] \xrightarrow{a} [a, 9, 8, b] \xrightarrow{b} [b, a, 9, 8] \quad \text{0 misses}$$

$$[\mathbf{e}, b, a, 9] \xrightarrow{8^*} [8, \mathbf{e}, b, a] \xrightarrow{9^*} [9, 8, \mathbf{e}, b] \xrightarrow{a^*} [a, 9, 8, \mathbf{e}] \xrightarrow{b^*} [b, a, 9, 8] \quad \text{4 misses}$$

- $|\mathrm{UCB}(s)| = 4$
- $|\mathrm{ECB}(s)| = 1$
- $n = 4$
- number of additional misses $= 4$

LRU: new computation of CRPD using ECB

- ECB derivation used only to know if the set is used by the preempting task:

$$\mathrm{CRPD}_{\mathrm{ECB}}^{\mathrm{LRU}}(s) = \begin{cases} 0 & \text{if } \mathrm{ECB}(s) = \emptyset \\ \mathrm{BRT} \cdot n & \text{otherwise} \end{cases}$$

CRPD for FIFO: Pitfalls

SAARLAND UNIVERSITY
COMPUTER SCIENCE

$$[b, a] \xrightarrow{a} [b, a] \xrightarrow{e^*} [e, b] \xrightarrow{b} [e, b] \xrightarrow{c^*} [c, e] \xrightarrow{e} [c, e] \qquad \text{2 misses}$$

CRPD for FIFO: Pitfalls

SAARLAND UNIVERSITY
COMPUTER SCIENCE

ECBs $= \{\mathbf{x}\}$

$$[b,a] \xrightarrow{a} [b,a] \xrightarrow{e^*} [e,b] \xrightarrow{b} [e,b] \xrightarrow{c^*} [c,e] \xrightarrow{e} [c,e] \quad \text{2 misses}$$

$$[\mathbf{x},b] \xrightarrow{a^*} [a,\mathbf{x}] \xrightarrow{e^*} [e,a] \xrightarrow{b^*} [b,e] \xrightarrow{c^*} [c,b] \xrightarrow{e^*} [e,c] \quad \text{5 misses}$$

- $|\mathrm{UCB}(s)| = 2$
- $|\mathrm{ECB}(s)| = 1$
- $n = 2$
- But: number of additional misses$= 3$

Definition – Relative Miss-Competitiveness

SAARLAND UNIVERSITY
COMPUTER SCIENCE

Notation

$m_P(p, s)$ = *number of misses that policy P incurs on access sequence $s \in M^*$ starting in state p*

Definition (Relative miss competitiveness)

Policy P is (k, c)-miss-competitive relative to policy Q, if

$$m_P(p, s) \leq k \cdot m_Q(q, s) + c$$

for all access sequences $s \in M^*$ and compatible cache-set states p, q.

Definition – Relative Miss-Competitiveness

Notation

$m_P(p, s) \quad = \quad$ *number of misses that policy P incurs on access sequence $s \in M^*$ starting in state p*

Definition (Relative miss competitiveness)

Policy P is (k, c)-miss-competitive relative to policy Q, if

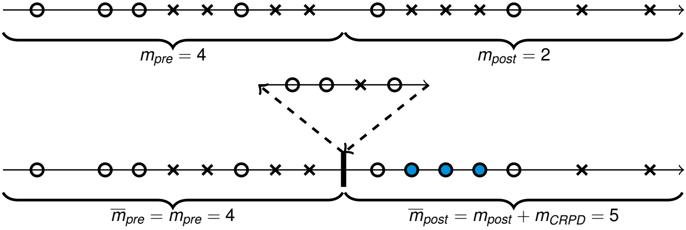
$$m_P(p, s) \leq k \cdot m_Q(q, s) + c$$

for all access sequences $s \in M^*$ and compatible cache-set states p, q.

- PLRU(n) is $(1, 0)$-miss-competitive relative to LRU($1 + log_2 n$).
- FIFO(n) is $(\frac{n}{n-r+1}, r)$-miss-competitive relative to LRU(r).

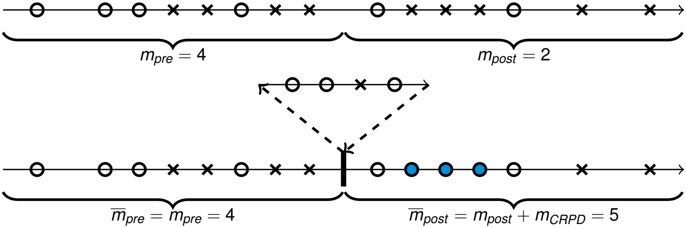
A sequence of memory accesses

- Notation:
  - m = number of misses
  - $\overline{m}$ = number of misses in the case of preemption



A sequence of memory accesses

- Notation:
  - m = number of misses
  - $\overline{m}$ = number of misses in the case of preemption



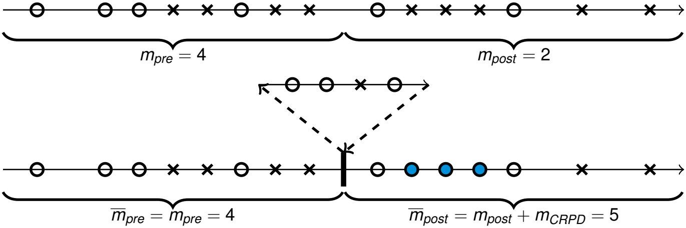
- Relative miss competitiveness:

$$\overline{m}^{(t)} = \overline{m}^{(t)}_{pre} + \overline{m}^{(t)}_{post}$$

A sequence of memory accesses

SAARLAND UNIVERSITY
COMPUTER SCIENCE

- Notation:
  - m = number of misses
  - $\overline{m}$ = number of misses in the case of preemption

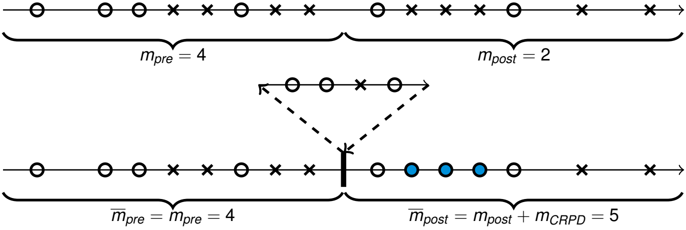


- Relative miss competitiveness:

$$\begin{aligned}\overline{m}^{(t)} &= \overline{m}^{(t)}_{pre} + \overline{m}^{(t)}_{post} \\ &\leq [k \cdot m^{\mathrm{LRU}(s)}_{pre} + c] + [k \cdot (m^{\mathrm{LRU}(s)}_{post} + m^{\mathrm{LRU}(s)}_{CRPD}) + c]\end{aligned}$$

A sequence of memory accesses

- Notation:
  - m = number of misses
  - $\overline{m}$ = number of misses in the case of preemption

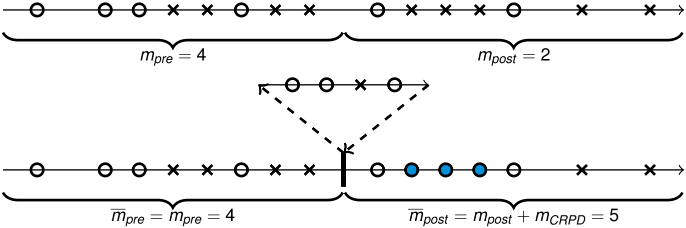


- Relative miss competitiveness:

$$
\begin{aligned}
\overline{m}^{(t)} &= \overline{m}^{(t)}_{pre} + \overline{m}^{(t)}_{post} \\
&\leq [k \cdot m^{\mathrm{LRU}(s)}_{pre} + c] + [k \cdot (m^{\mathrm{LRU}(s)}_{post} + m^{\mathrm{LRU}(s)}_{CRPD}) + c] \\
&= [k \cdot (m^{\mathrm{LRU}(s)}_{pre} + m^{\mathrm{LRU}(s)}_{post}) + c] + [k \cdot m^{\mathrm{LRU}(s)}_{CRPD} + c]
\end{aligned}
$$

A sequence of memory accesses

- Notation:
  - m = number of misses
  - $\overline{m}$ = number of misses in the case of preemption



- Relative miss competitiveness:

$$
\begin{aligned}
\overline{m}^{(t)} &= \overline{m}^{(t)}_{pre} + \overline{m}^{(t)}_{post} \\
&\leq [k \cdot m^{\mathrm{LRU}(s)}_{pre} + c] + [k \cdot (m^{\mathrm{LRU}(s)}_{post} + m^{\mathrm{LRU}(s)}_{CRPD}) + c] \\
&= [k \cdot (m^{\mathrm{LRU}(s)}_{pre} + m^{\mathrm{LRU}(s)}_{post}) + c] + [k \cdot m^{\mathrm{LRU}(s)}_{CRPD} + c] \\
&= [k \cdot m^{\mathrm{LRU}(s)} + c] + [k \cdot m^{\mathrm{LRU}(s)}_{CRPD} + c]
\end{aligned}
$$

Relative Competitiveness – Application

- PLRU(8) using LRU(4):

$$\overline{m}^{\mathrm{PLRU}(8)} \leq m^{\mathrm{LRU}(4)} + m_{CRPD}^{\mathrm{LRU}(4)}$$

- FIFO(8) using LRU(5):

$$\overline{m}^{\mathrm{FIFO}(8)} \leq (2 \cdot m^{\mathrm{LRU}(5)} + 5) + (2 \cdot m_{CRPD}^{\mathrm{LRU}(5)} + 5)$$

Conclusions

SAARLAND UNIVERSITY
COMPUTER SCIENCE

Pitfalls

- *LRU:*
  $|ECBs|$ is not an upper-bound
- *FIFO and PLRU:*
  $|UCBs|$, $|ECBs|$ and n do not bound the number of additional misses

Solutions

- *LRU:*
  $|ECBs| = 0$
  $\Rightarrow$ the set is not used
- *FIFO and PLRU:*
  using relative competitiveness and LRU bounds

Further reading

- Altmeyer, S. and Burguière, C. (2009).
  A New Notion of Useful Cache Block to Improve the Bounds of Cache-Related Preemption Delay.
  In ECRTS '09 pp. 109–118, IEEE Computer Society.
- Lee, C.-G., Hahn, J., Min, S. L., Ha, R., Hong, S., Park, C. Y., Lee, M. and Kim, C. S. (1996).
  Analysis of cache-related preemption delay in fixed-priority preemptive scheduling.
  In RTSS'96 p. 264, IEEE Computer Society.
- Negi, H. S., Mitra, T. and Roychoudhury, A. (2003).
  Accurate estimation of cache-related preemption delay.
  In CODES+ISSS'03 ACM.
- Reineke, J. (2008).
  Caches in WCET Analysis.
  PhD thesis, Universität des Saarlandes, Saarbrücken.
- Staschulat, J. and Ernst, R. (2007).
  Scalable precision cache analysis for real-time software.
  ACM TECS *6*, 25.
- Tan, Y. and Mooney, V. (2004).
  Integrated intra- and inter-task cache analysis for preemptive multi-tasking real-time systems.
  In SCOPES'04 pp. 182–199,.
- Tomiyama, H. and Dutt, N. D. (2000).

Upper-bound on the CRPD - direct-mapped caches

- using UCB [Lee et al., 1996]:

$$\text{CRPD}_{\text{UCB}} = \text{BRT} \cdot |\{s_i \mid \exists m \in \text{UCB} : m \bmod c = s_i\}|$$

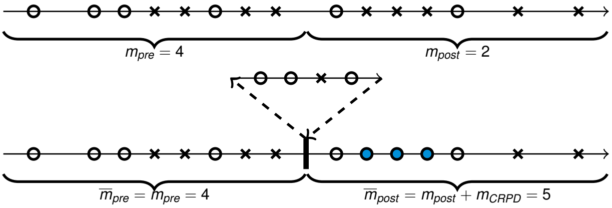
- using ECB [Tomiyama and Dutt, 2000]:

$$\text{CRPD}_{\text{ECB}} = \text{BRT} \cdot |\{s_i \mid \exists m \in \text{ECB} : m \bmod c = s_i\}|$$

- using UCB and ECB [Negi et al., 2003, Tan and Mooney, 2004]:

$$\begin{aligned}\text{CRPD}_{\text{UCB\&ECB}} = \text{BRT} \cdot |\{s_i \mid\ & \exists m \in \text{UCB} : m \bmod c = s_i \\ & \wedge \exists m' \in \text{ECB} : m' \bmod c = s_i\}|\end{aligned}$$

Relative Competitiveness – Example



$$\begin{aligned}\overline{m}^{(t)}_{pre} &\leq k \cdot \overline{m}^{\mathrm{LRU}(s)}_{pre} + c\\ &= k \cdot m^{\mathrm{LRU}(s)}_{pre} + c\end{aligned}$$

$$\begin{aligned}\overline{m}^{(t)}_{post} &\leq k \cdot \overline{m}^{\mathrm{LRU}(s)}_{post} + c\\ &= k \cdot (m^{\mathrm{LRU}(s)}_{post} + m^{\mathrm{LRU}(s)}_{CRPD}) + c\end{aligned}$$

$$\begin{aligned}\overline{m}^{(t)} &= \overline{m}^{(t)}_{pre} + \overline{m}^{(t)}_{post}\\ &\leq k \cdot m^{\mathrm{LRU}(s)}_{pre} + c + k \cdot (m^{\mathrm{LRU}(s)}_{post} + m^{\mathrm{LRU}(s)}_{CRPD}) + c\\ &= (k \cdot (m^{\mathrm{LRU}(s)}_{pre} + m^{\mathrm{LRU}(s)}_{post}) + c) + (k \cdot m^{\mathrm{LRU}(s)}_{CRPD} + c)\\ &= (k \cdot m^{\mathrm{LRU}(s)} + c) + (k \cdot m^{\mathrm{LRU}(s)}_{CRPD} + c).\end{aligned}$$

Relative Competitiveness – Application

- PLRU(n) using LRU($1 + log_2 n$):
  - $k = 1, c = 0$

$$\overline{m}^{\text{PLRU}(n)} \leq m^{\text{LRU}(1+log_2 n)} + m_{CRPD}^{\text{LRU}(1+log_2 n)}$$

  - Example (n=8):

$$\overline{m}^{\text{PLRU}(8)} \leq m^{\text{LRU}(4)} + m_{CRPD}^{\text{LRU}(4)}$$

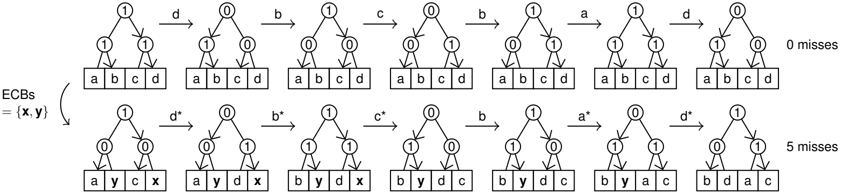
- FIFO(n) using LRU(r):
  - $k = \frac{n}{n-r+1}, c = r$

$$\overline{m}^{\text{FIFO}(n)} \leq (\frac{n}{n-r+1} \cdot m^{\text{LRU}(r)} + r) + (\frac{n}{n-r+1} \cdot m_{CRPD}^{\text{LRU}(r)} + r)$$

  - Example (n=8,r=5):

$$\overline{m}^{\text{FIFO}(8)} \leq (2 \cdot m^{\text{LRU}(5)} + 5) + (2 \cdot m_{CRPD}^{\text{LRU}(5)} + 5)$$

CRPD for PLRU: Pitfalls



- $|\mathrm{UCB}(s)| = 4$
- $|\mathrm{ECB}(s)| = 2$
- $n = 4$
- But: number of additional misses$= 5$