EXPLORING PREDICTIVE REPLACEMENT POLICIES FOR INSTRUCTION CACHE AND BRANCH TARGET BUFFER

Samira Mirbagher Ajorpaz
Elba Garza
Sangam Jindal
Daniel A. Jiménez
Exploring Predictive Replacement Policies for Instruction Cache and Branch Target Buffer

Samira Mirbagher Ajorpaz
Elba Garza
Sangam Jindal
Daniel A. Jiménez
EXPLORING PREDICTIVE REPLACEMENT POLICIES FOR INSTRUCTION CACHE AND BRANCH TARGET BUFFER

Samira Mirbagher Ajorpaz
Elba Garza
Sangam Jindal
Daniel A. Jiménez

Dead
EXPLORING PREDICTIVE REPLACEMENT POLICIES FOR INSTRUCTION CACHE AND BRANCH TARGET BUFFER

Samira Mirbagher Ajorpaz
Elba Garza
Sangam Jindal
Daniel A. Jiménez
Thousands of workloads from popular benchmark suites

Part of fifth Championship Branch Prediction, provided by Samsung

223 training + 439 evaluation workloads
Many applications have significant I-cache and BTB misses
pipeline flush

wrong speculation

branch misspred

Direction Miss
target miss

ITLB Miss
Icache Miss
branch resteer

fetch latency

front end bound

fetch bandwidth

back end bound

core bound

memory bound

src1 src2 DIV
exec port mem L3 L2 L1 store
pipeline flush

wrong speculation

branch misspred

front end bound

fetch latency

ITLB Miss

Icache miss

branch resteer

stalled

core bound

memory bound

back end bound

fetch bandwidth

src1

src2

store

divider

exec port

mem

L3

L2

L1

store
No previous work on
Predictive Replacement Policies for I-cache and BTB
If A becomes dead,
B and C are likely to become dead too.
Sampling Dead Block Prediction learns from a small number of sets.
Sampling Dead Block Prediction reduces many dead blocks in LL cache
SDBP increase I-cache MPKI by 4% in average
In I-cache or BTB, one PC accesses only one set
GhRp
GHRP correlates

Reuse behavior with control flow History

Global History

XOR

$PC_t$

 Signature
GHRP correlates Reuse behavior with control flow History

<table>
<thead>
<tr>
<th>$PC_{t-4}$</th>
<th>0</th>
<th>$PC_{t-3}$</th>
<th>0</th>
<th>$PC_{t-2}$</th>
<th>0</th>
<th>$PC_{t-1}$</th>
<th>0</th>
</tr>
</thead>
</table>

XOR

$PC_t$

↓

Signature
Extra information kept in I-cache block
GHRP prediction is done by tracking the behavior using the signature.
Extra information kept in I-cache block
Voting is required for GHRP decisions
Voting is required for GHRP decisions

Hash1 > Threshold → Hash2 > Threshold → Hash3 > Threshold → Prediction
Voting is required for GHRP decisions

Hash1 > Threshold

Hash2 > Threshold

Hash3 > Threshold

Majority vote

Prediction
GHRP correlates
Reuse behavior with control flow History

\[
\begin{array}{cccccc}
PC_{t-4} & 0 & PC_{t-3} & 0 & PC_{t-2} & 0 & PC_{t-1} & 0 \\
\end{array}
\]

XOR

\[
PC_t
\]

\[
\Rightarrow
\]

Signature

GHRP prediction is done by tracking the behavior using the signature

\[
\uparrow
\]

Eviction

\[
\downarrow
\]

Reuse

Voting is required for GHRP decisions

\[
\text{Hash1} \quad \text{Hash2} \quad > \text{Threshold} \quad > \text{Threshold} \quad > \text{Threshold}
\]

\[
\text{Majority vote}
\]

\[
\downarrow
\]

Prediction

<table>
<thead>
<tr>
<th>valid</th>
<th>prediction</th>
<th>LRU stack</th>
<th>signature</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 bit</td>
<td>1 bit</td>
<td>3 bits</td>
<td>16 bits</td>
</tr>
</tbody>
</table>
New Signature
<table>
<thead>
<tr>
<th>valid</th>
<th>prediction</th>
<th>LRU stack</th>
<th>signature</th>
</tr>
</thead>
</table>

Victim Block

Miss
Not Bypass
Victim Block

Miss
Not Bypass
Victim Block

Miss
Not Bypass

New Block
<table>
<thead>
<tr>
<th>valid</th>
<th>prediction</th>
<th>LRU stack</th>
<th>signature</th>
</tr>
</thead>
</table>

Hit Block

Hit
valid  prediction  LRU stack  signature

Hit Block

Hit
Hit Block

Reuse

Hit

Hit Block
<table>
<thead>
<tr>
<th>( PC_{t-4} )</th>
<th>0</th>
<th>( PC_{t-3} )</th>
<th>0</th>
<th>( PC_{t-2} )</th>
<th>0</th>
<th>( PC_{t-1} )</th>
<th>0</th>
</tr>
</thead>
</table>

<< Shift Left >>

<table>
<thead>
<tr>
<th>( PC_{t-3} )</th>
<th>0</th>
<th>( PC_{t-2} )</th>
<th>0</th>
<th>( PC_{t-1} )</th>
<th>0</th>
<th>( PC_t )</th>
<th>0</th>
</tr>
</thead>
</table>

↓

New Global History
If A becomes dead in I-cache, B is likely to become dead in BTB too.
BTB and I-cache can share prediction resources.
BTB and I-cache can share prediction resources
BTB and I-cache
joint design

| $br_{t-4}$ | 0 | $br_{t-3}$ | 0 | $br_{t-2}$ | 0 | $br_{t-1}$ | 0 |
Workloads
- 662 traces
- Short-Mobile, Long-Mobile
- Short-Server, Long-Server

Branch Predictor
- Hashed Perceptron

Simulator
- CBP5
- Trace driven
- MPKI

Comparison
- LRU (baseline)
- Random
- SRRIP
- SDBP

Workloads
- 662 traces
- Short-Mobile, Long-Mobile
- Short-Server, Long-Server
- CBP5, Samsung

I-cache
- 64KB
- 8 Way
- 64B

BTB
- 4K Entry
- 8 Way
**Simulator**
- CBP5
- Trace driven
- MPKI

**Branch Predictor**
- Hashed Perceptron

**Workloads**
- 662 traces
- Short-Mobile, Long-Mobile, Short-Server, Long-Server
- CBP5, Samsung

**Comparison**
- LRU (baseline)
- Random
- SRRIP
- SDBP

**I-cache**
- 64KB
- 8 Way
- 64B

**BTB**
- 4K Entry
- 8 Way
Simulator
- CBP5
- Trace driven
- MPKI

Comparison
- LRU (baseline)
- Random
- SRRIP
- SDBP

Branch Predictor
- Hashed Perceptron

Workloads
- 662 traces
- Short-Mobile, Long-Mobile, Short-Server, Long-Server
- CBP5, Samsung

I-cache
- 64KB
- 8 Way
- 64B

BTB
- 4K Entry
- 8 Way
Branch Predictor

Hashed Perceptron

Simulator

Trace driven

Workloads

662 traces

CBP5, Samsung

Trace driven

LRU (baseline)

Comparison

SDBP

SRRIP

Random

64KB 8 Way

4K Entry

TB

I-cache

8 Way 64KB

Branch Predictor

Hashed

Perceptron
Branch Predictor
Hashed Perceptron

Simulator
CBP5
Trace driven
MPKI

Comparison
LRU(baseline)
Random
SRRIP
SDBP

Workloads
662 traces
Short-Mobile,
Long-Mobile,
Short-Server,
Long-Server

CBP5,
Samsung

I-cache
64KB
8 Way
64B

BTB
4K Entry
8 Way
### 64KB, 8-way I-cache with 64B blocks

<table>
<thead>
<tr>
<th>Component</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 bit counters</td>
<td>16 bits</td>
<td>4,096 entries</td>
</tr>
<tr>
<td>4,096 entries</td>
<td>1,024</td>
<td>blocks</td>
</tr>
<tr>
<td>Prediction Tables</td>
<td>3KB</td>
<td></td>
</tr>
<tr>
<td>Signature bits</td>
<td>2KB</td>
<td>1,024 blocks</td>
</tr>
<tr>
<td>Prediction bits</td>
<td>128B</td>
<td></td>
</tr>
<tr>
<td>History Register</td>
<td>2B</td>
<td>16 bits</td>
</tr>
</tbody>
</table>

**Total storage overhead GHRP is 5.13KB or 8% of the capacity of the I-cache**
With 95% certainty GHRP reduces I-cache MPKI by 33% compared to LRU.
With 95% certainty GHRP reduces BTB MPKI by 41% compared to LRU
Questions?