Wormhole branch prediction using multi-dimensional histories

Jorge Albericio,
Joshua San Miguel,
Natalie Enright Jerger, and Andreas Moshovos


14
JEWELRY
WEBB'S SHOES
WEBB'S SHOES
1 | 4

Motivation

foreach frame:
  foreach object:
    if distance(**object**, **p**) < threshold:
    { /* do something */ }

Motivation

```
foreach frame:
    foreach object:
        if  distance(object, p) < threshold:
        { /* do something */ }
```

Motivation

```
foreach frame:
    foreach object:
        if  distance(object, p) < threshold:
        { /* do something */ }
```

Motivation

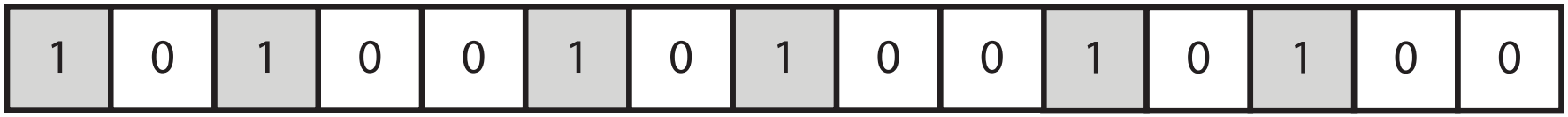
foreach frame:

  foreach object:

    if distance(**object**, **p**) < threshold:

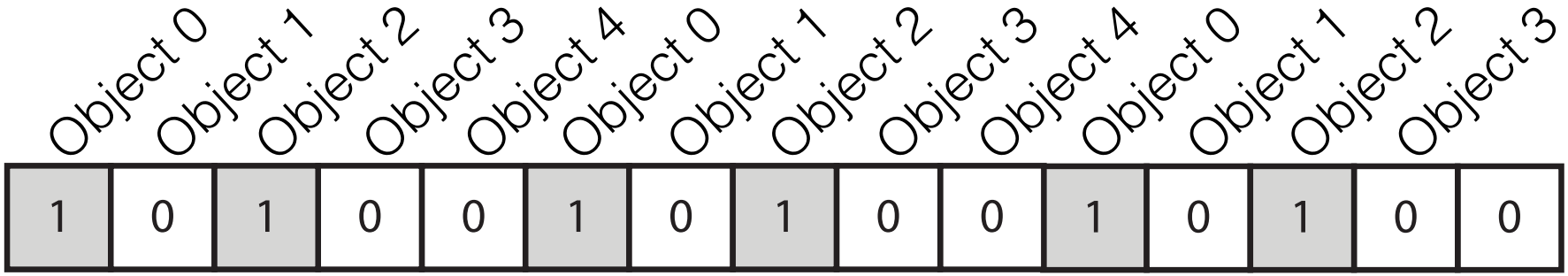
    { /* do something */ }

Multidimensional histories



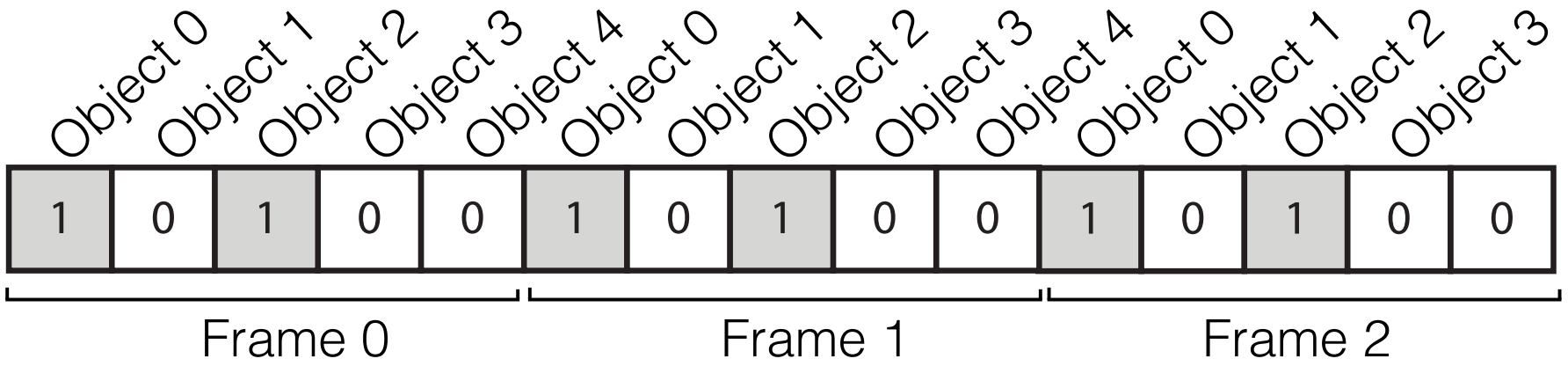

```
foreach frame:
  foreach object:
    if  distance(object, p) < threshold:
    { /* do something */ }
```

Multidimensional histories



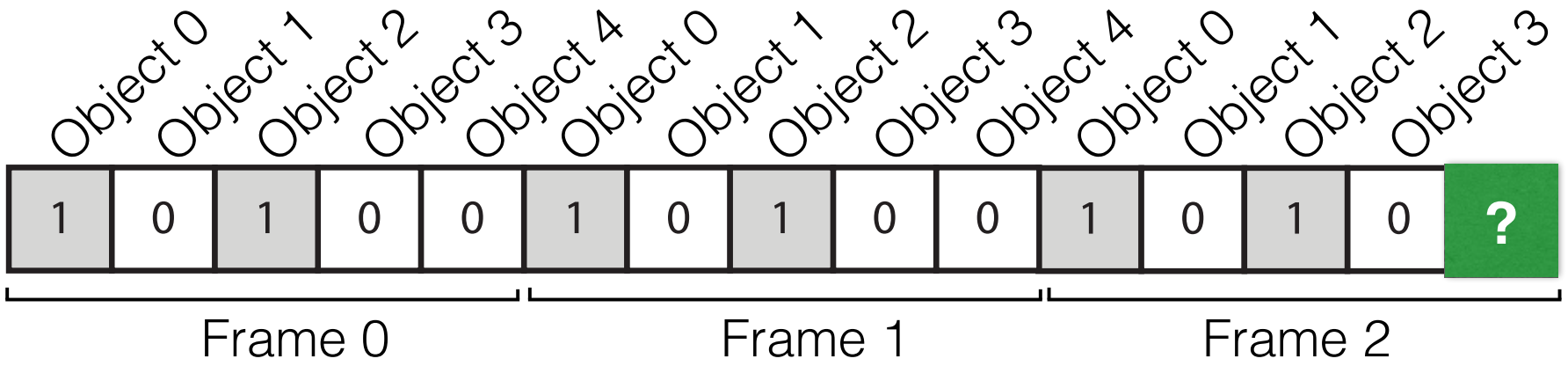

```
foreach frame:
  foreach object:
    if  distance(object, p) < threshold:
    { /* do something */ }
```

Multidimensional histories



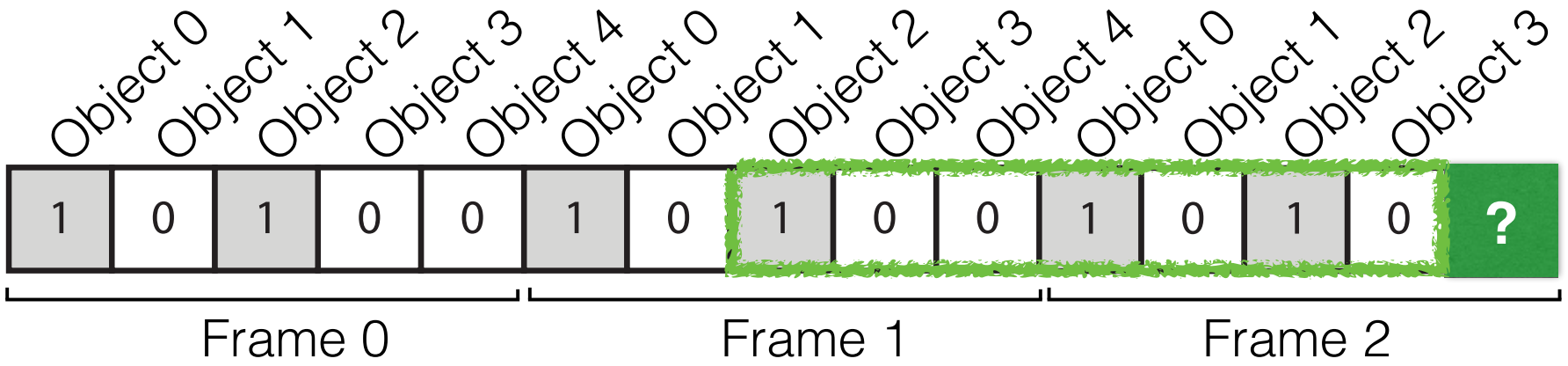

```
foreach frame:
  foreach object:
    if  distance(object, p) < threshold:
    { /* do something */ }
```

Multidimensional histories



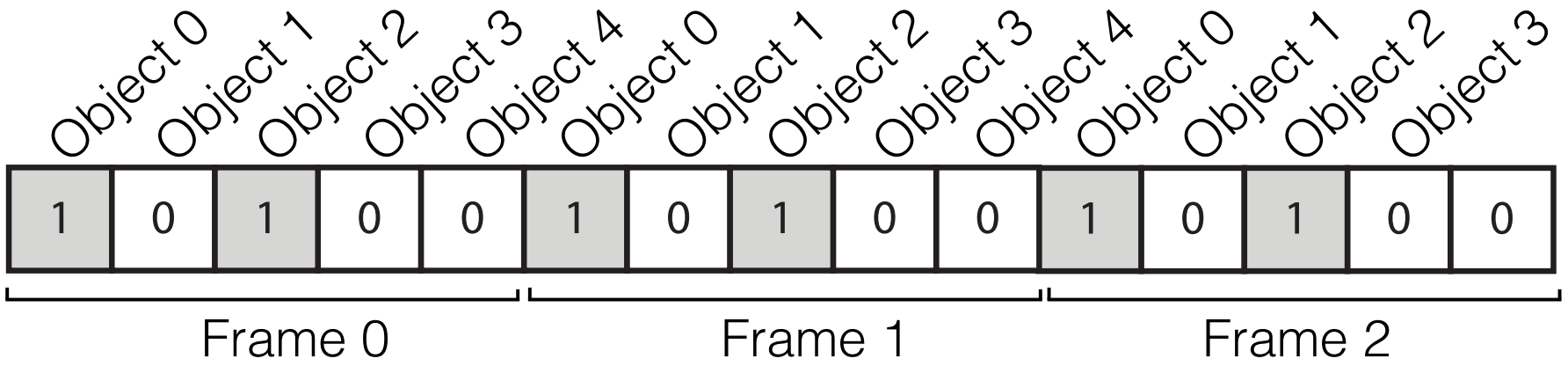

```
foreach frame:
  foreach object:
    if  distance(object, p) < threshold:
    { /* do something */ }
```

Multidimensional histories



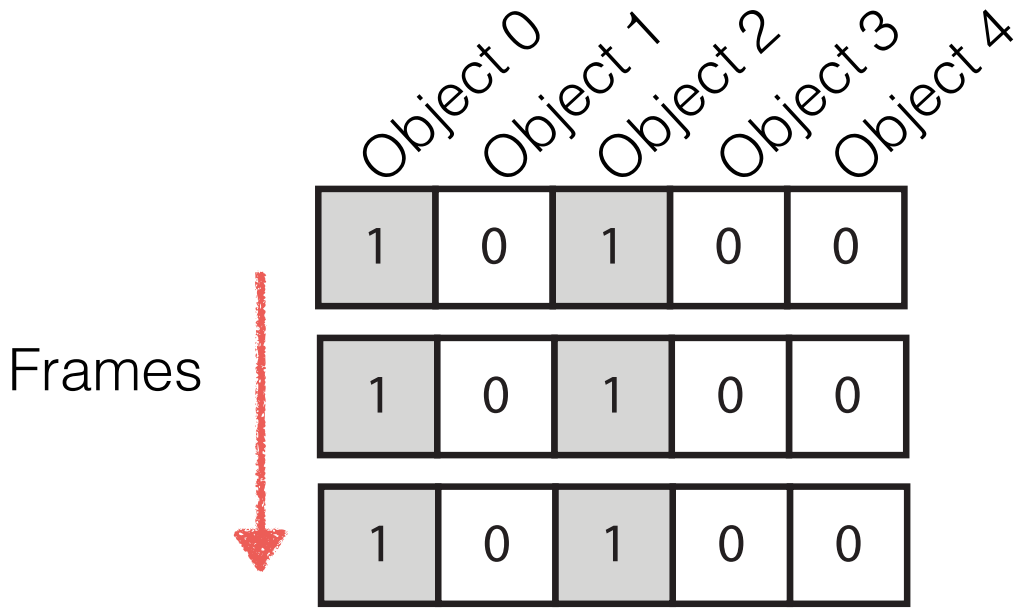

```
foreach frame:
  foreach object:
    if  distance(object, p) < threshold:
    { /* do something */ }
```

Multidimensional histories



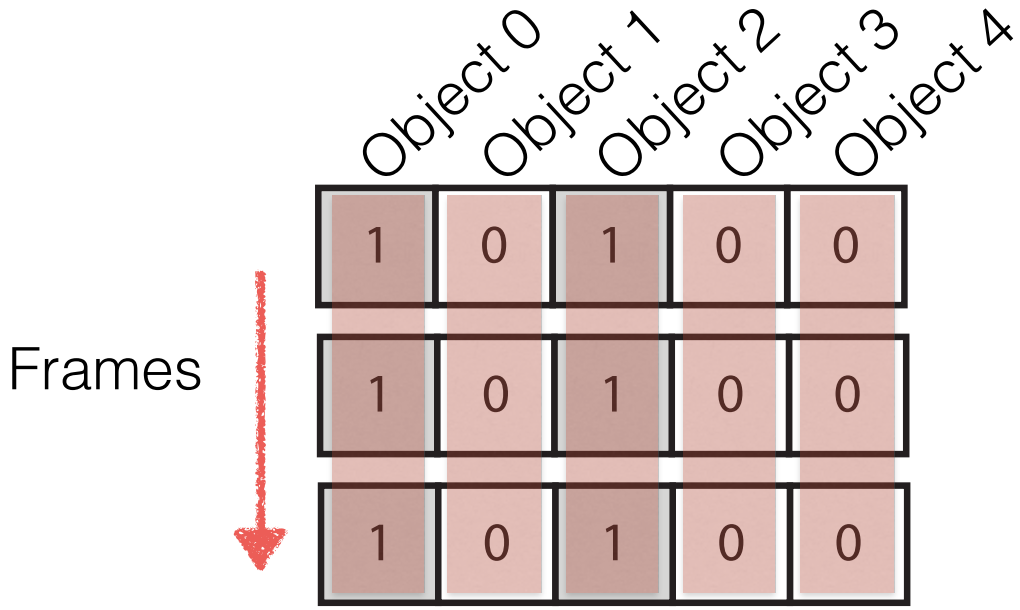

```
foreach frame:
  foreach object:
    if  distance(object, p) < threshold:
    { /* do something */ }
```

Multidimensional histories




```
foreach frame:
  foreach object:
    if  distance(object, p) < threshold:
    { /* do something */ }
```

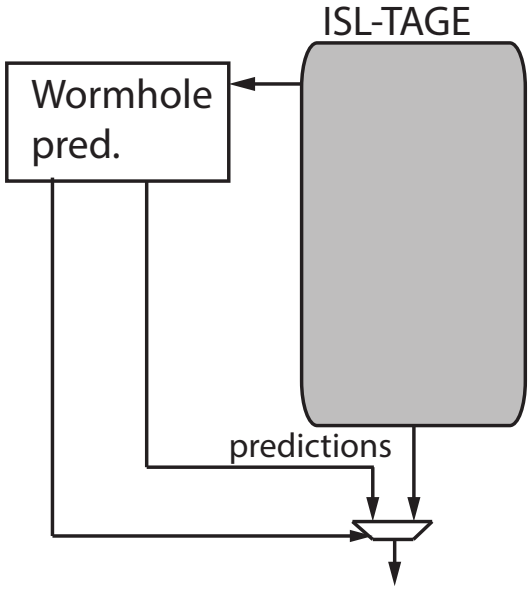
Multidimensional histories




```
foreach frame:
  foreach object:
    if  distance(object, p) < threshold:
    { /* do something */ }
```

Summary

- **Wormhole predictor**: Side-predictor on top of ISL-TAGE
- Able to capture columns, diagonals, and other patterns
- **3.995**, **2.498**, and **2.014** MPKI in the 4KB, 32KB, and unlimited competition tracks



Detecting diagonals

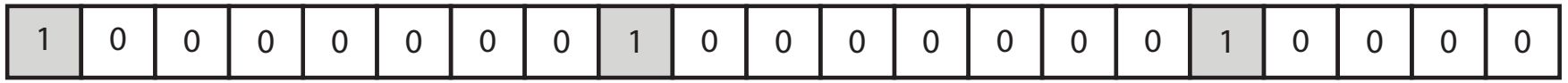
```
for j in N:
    for k in N:
        if  j != k:
            /*Calculation*/
```

Detecting diagonals

```
for j in N:
    for k in N:
        if  j != k:
            /*Calculation*/
```

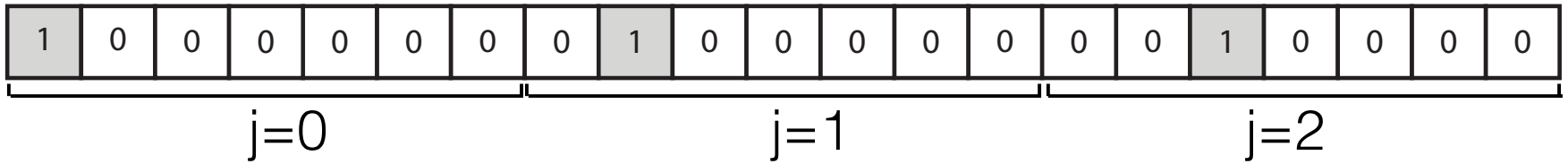
Detecting diagonals

```
for j in N:
    for k in N:
        if  j != k:
            /*Calculation*/
```

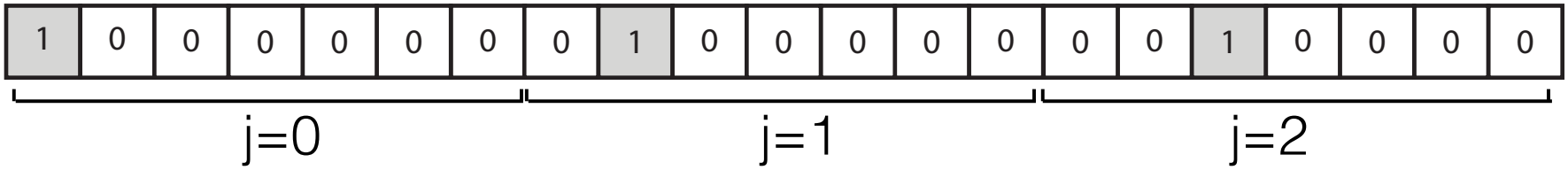


Detecting diagonals

```
for j in N:
    for k in N:
        if  j != k:
            /*Calculation*/
```

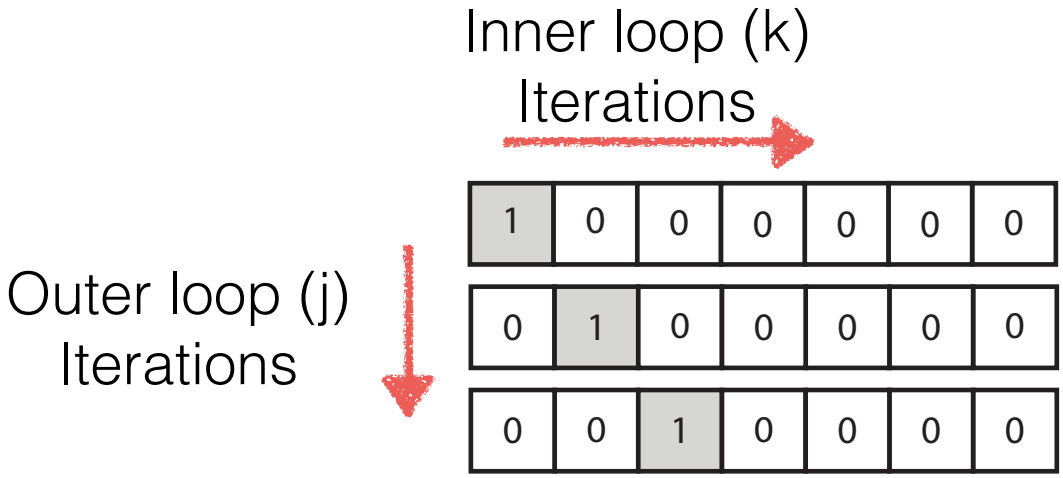


Detecting diagonals



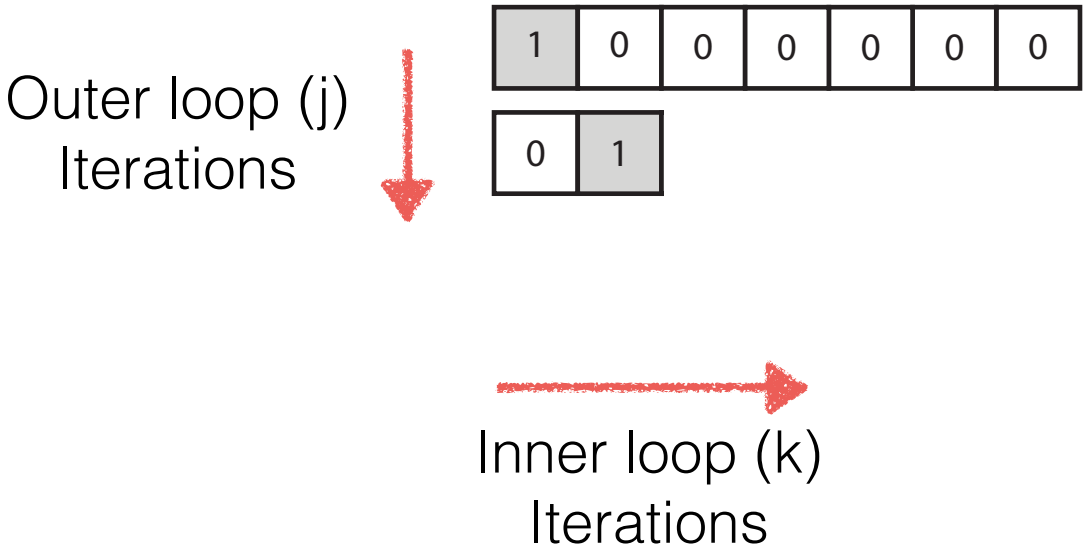

```
for j in N:          //Loop 1
   for k in N:       //Loop 2
      if  j != k:
          /*Calculation*/
```

Detecting diagonals



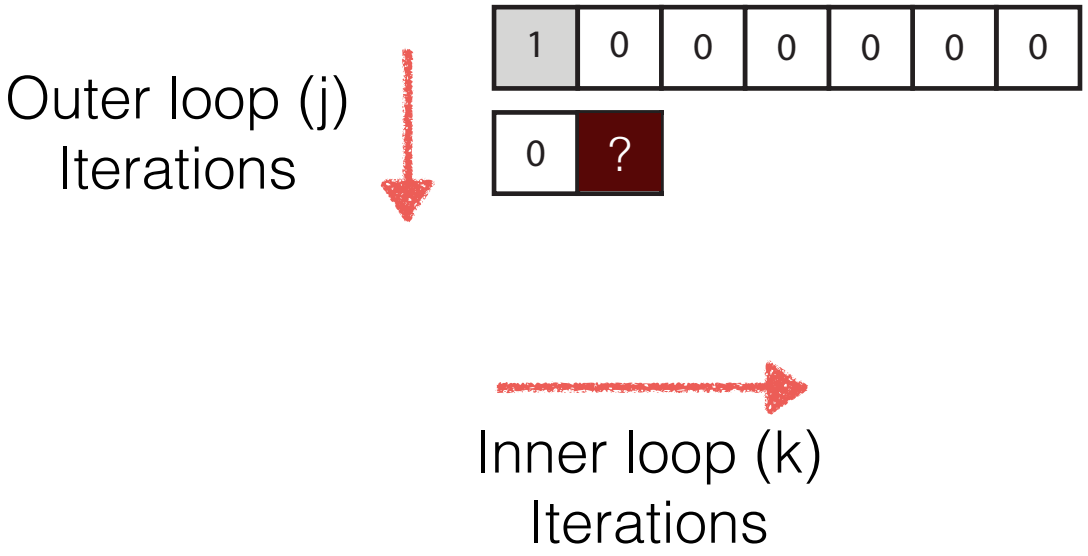

```
for j in N:          //Loop 1
    for k in N:      //Loop 2
        if  j != k:
            /*Calculation*/
```

Multidimensional histories



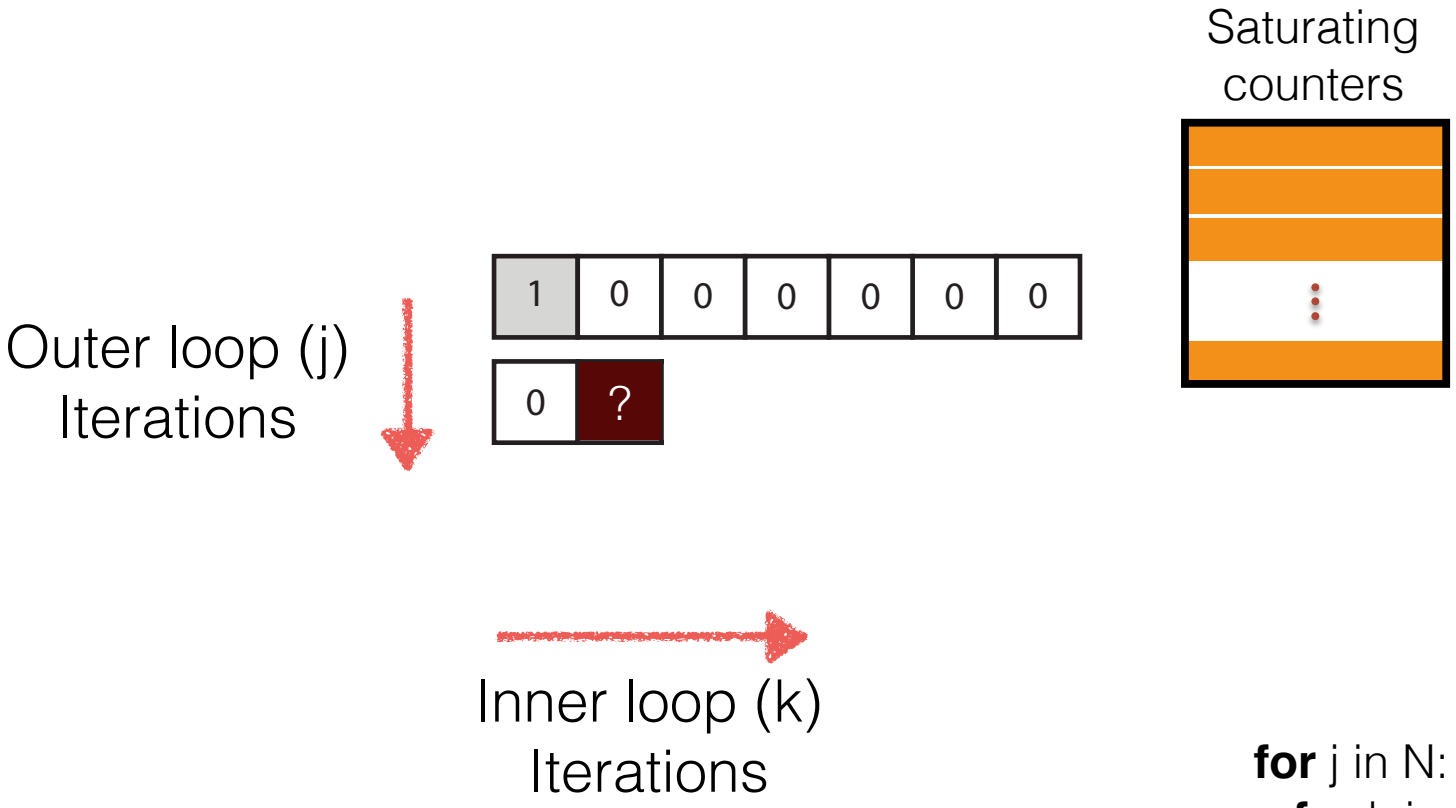

```
for j in N:
   for k in N:
      if  j != k:
         /*Calculation*/
```

Multidimensional histories



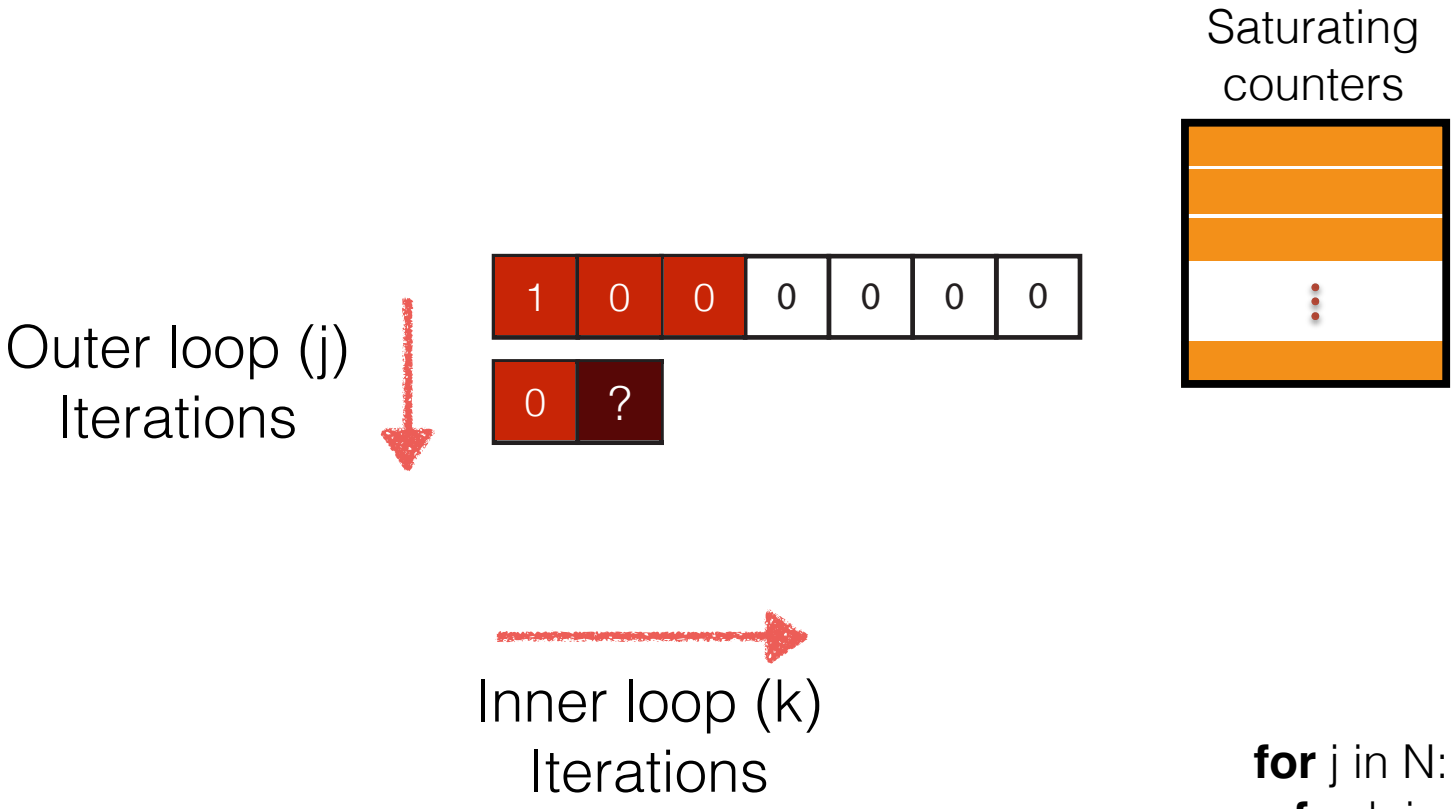

```
for j in N:
    for k in N:
        if  j != k:
            /*Calculation*/
```

Multidimensional histories



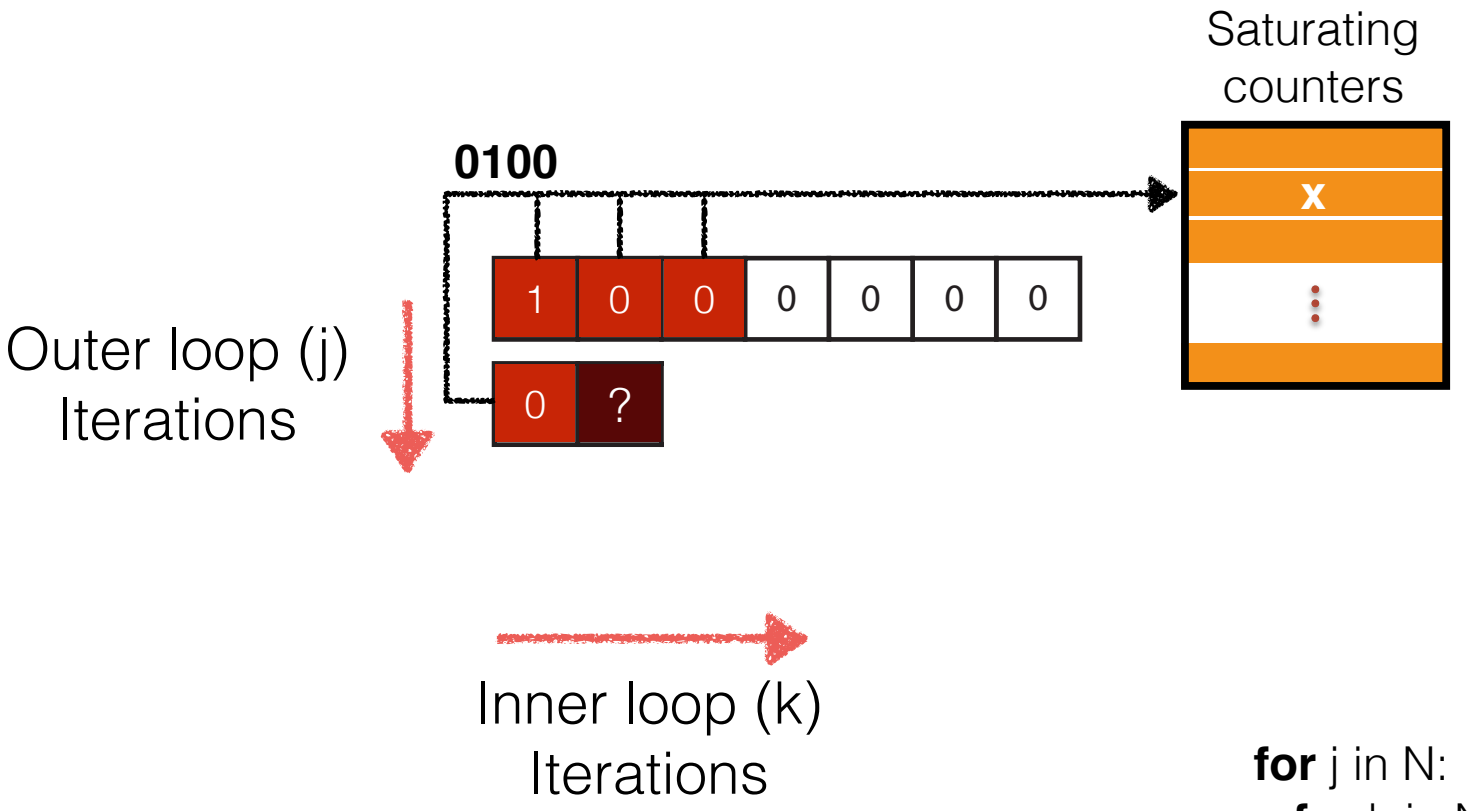

```
for j in N:
    for k in N:
        if  j != k:
            /*Calculation*/
```

Multidimensional histories



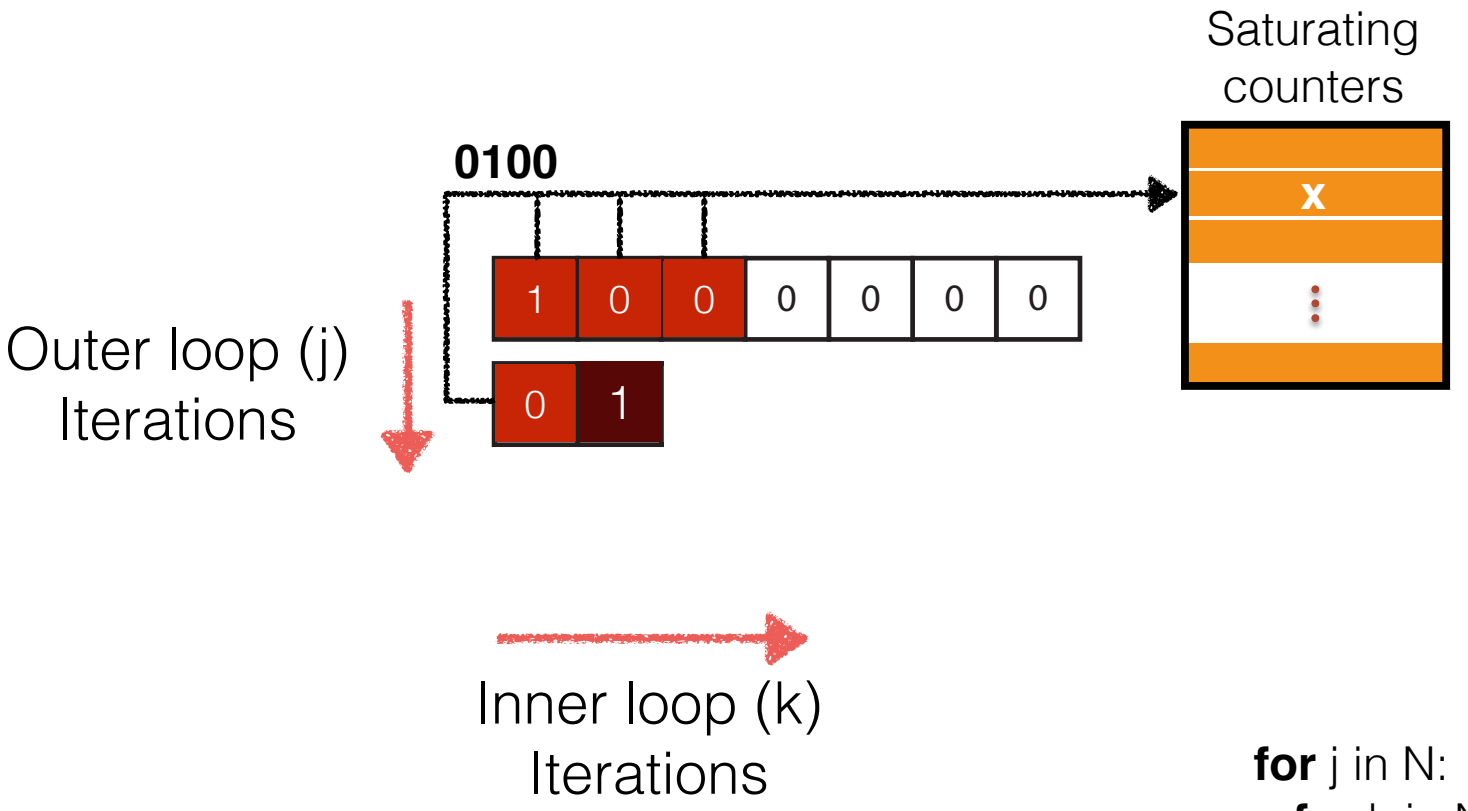

```
for j in N:
    for k in N:
        if  j != k:
            /*Calculation*/
```

Multidimensional histories



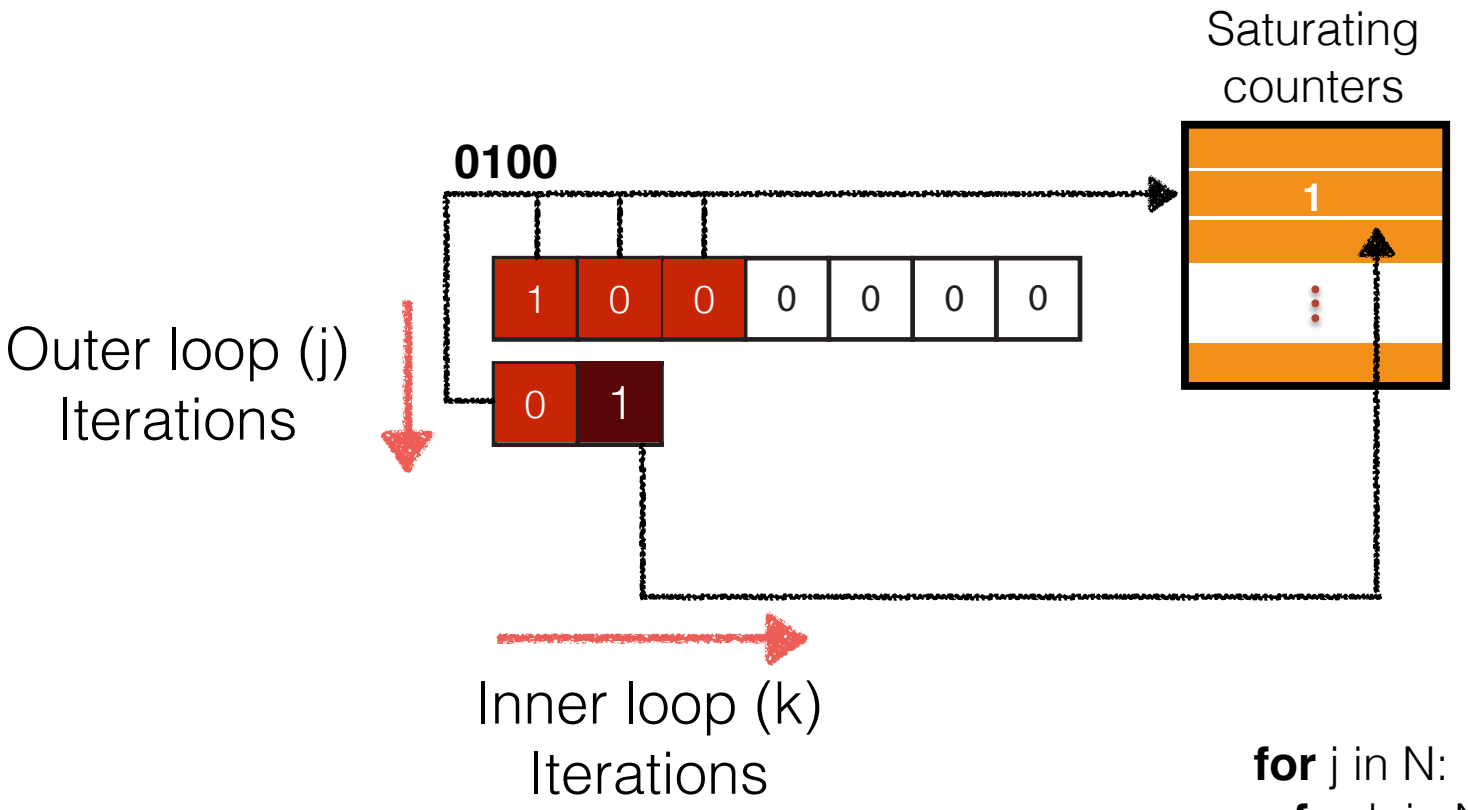

```
for j in N:
    for k in N:
        if  j != k:
            /*Calculation*/
```

Multidimensional histories



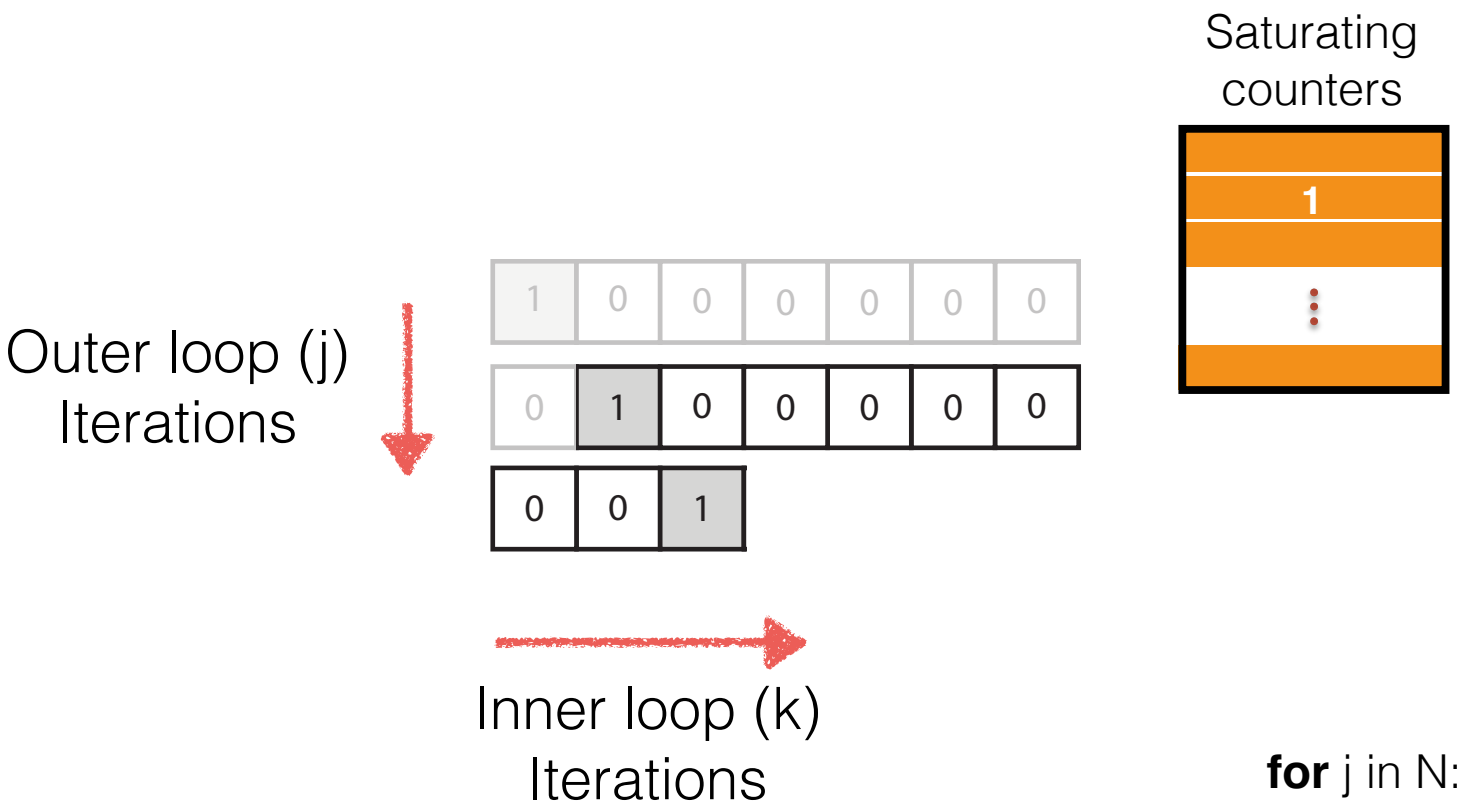

```
for j in N:
    for k in N:
        if  j != k:
            /*Calculation*/
```

Multidimensional histories



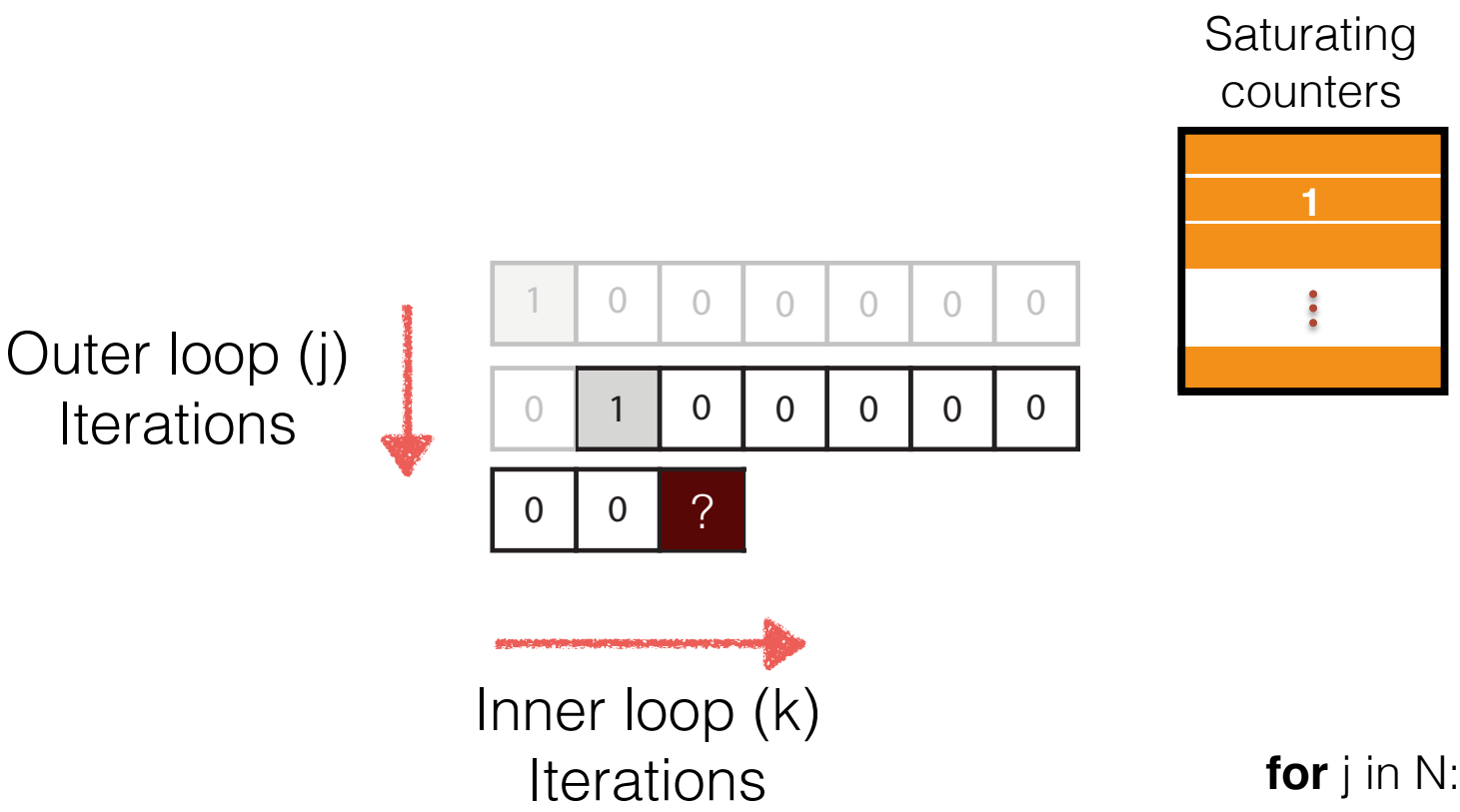

```
for j in N:
    for k in N:
        if  j != k:
            /*Calculation*/
```

Multidimensional histories



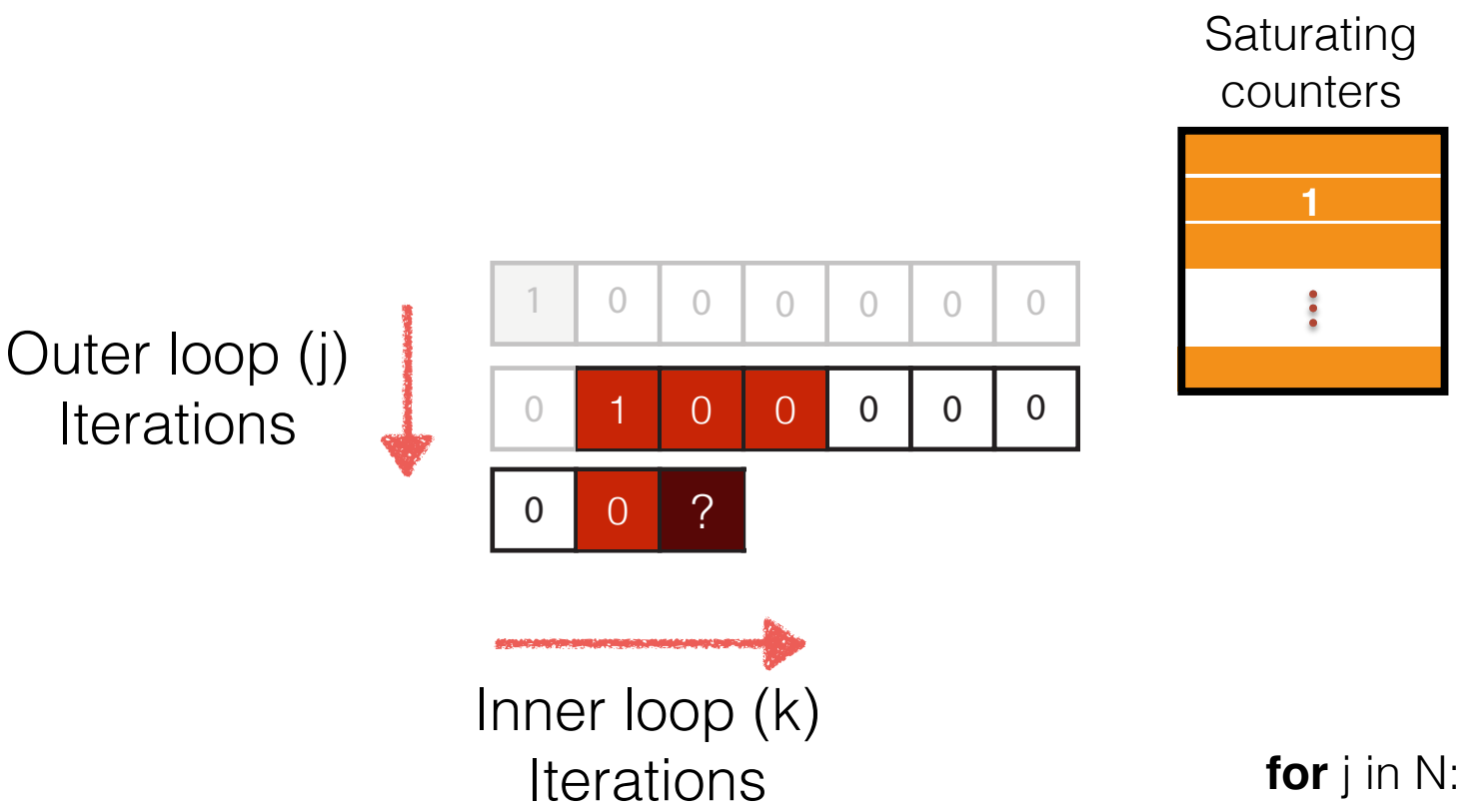

```
for j in N:
    for k in N:
        if  j != k:
            /*Calculation*/
```

Multidimensional histories



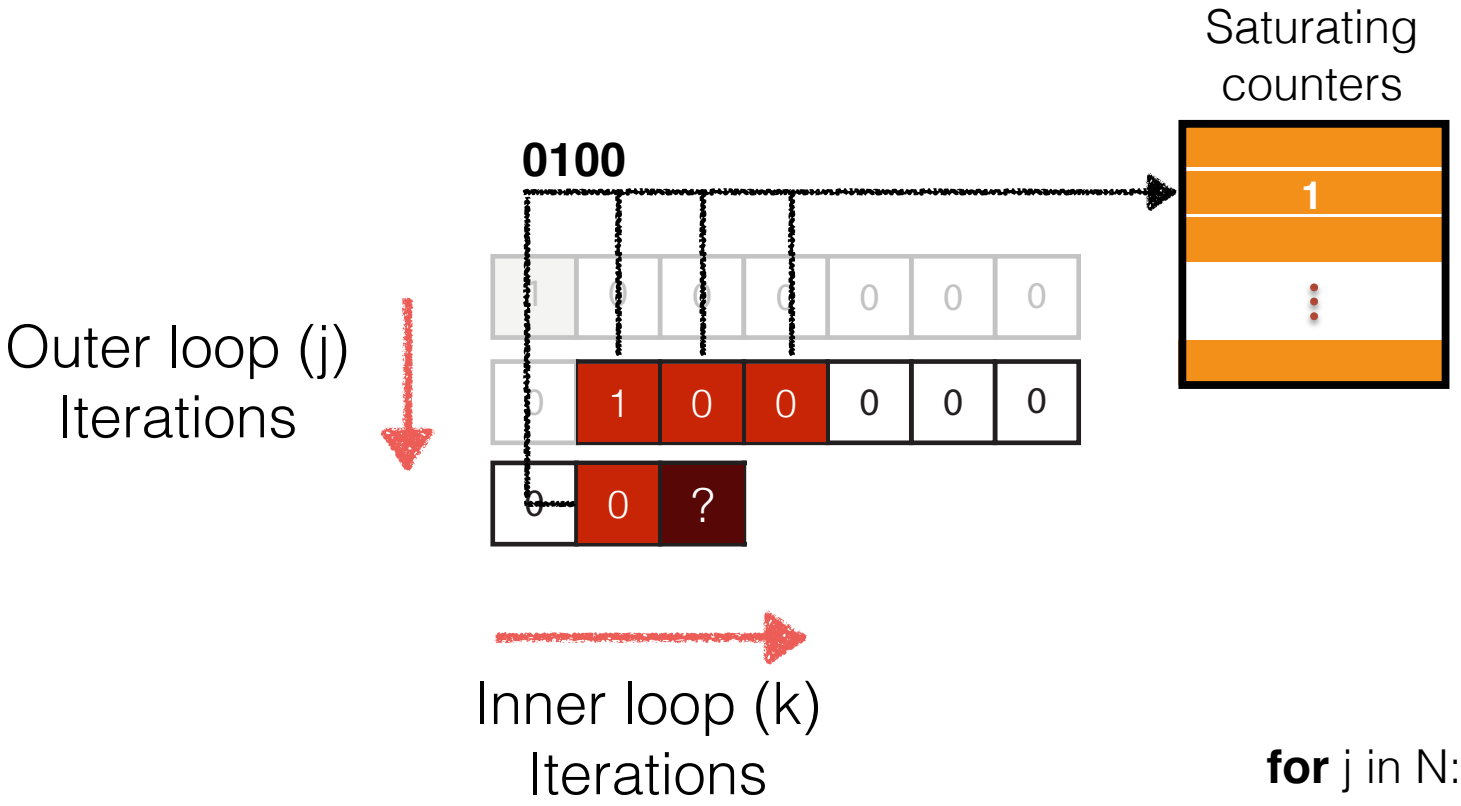

```
for j in N:
   for k in N:
      if  j != k:
         /*Calculation*/
```

Multidimensional histories



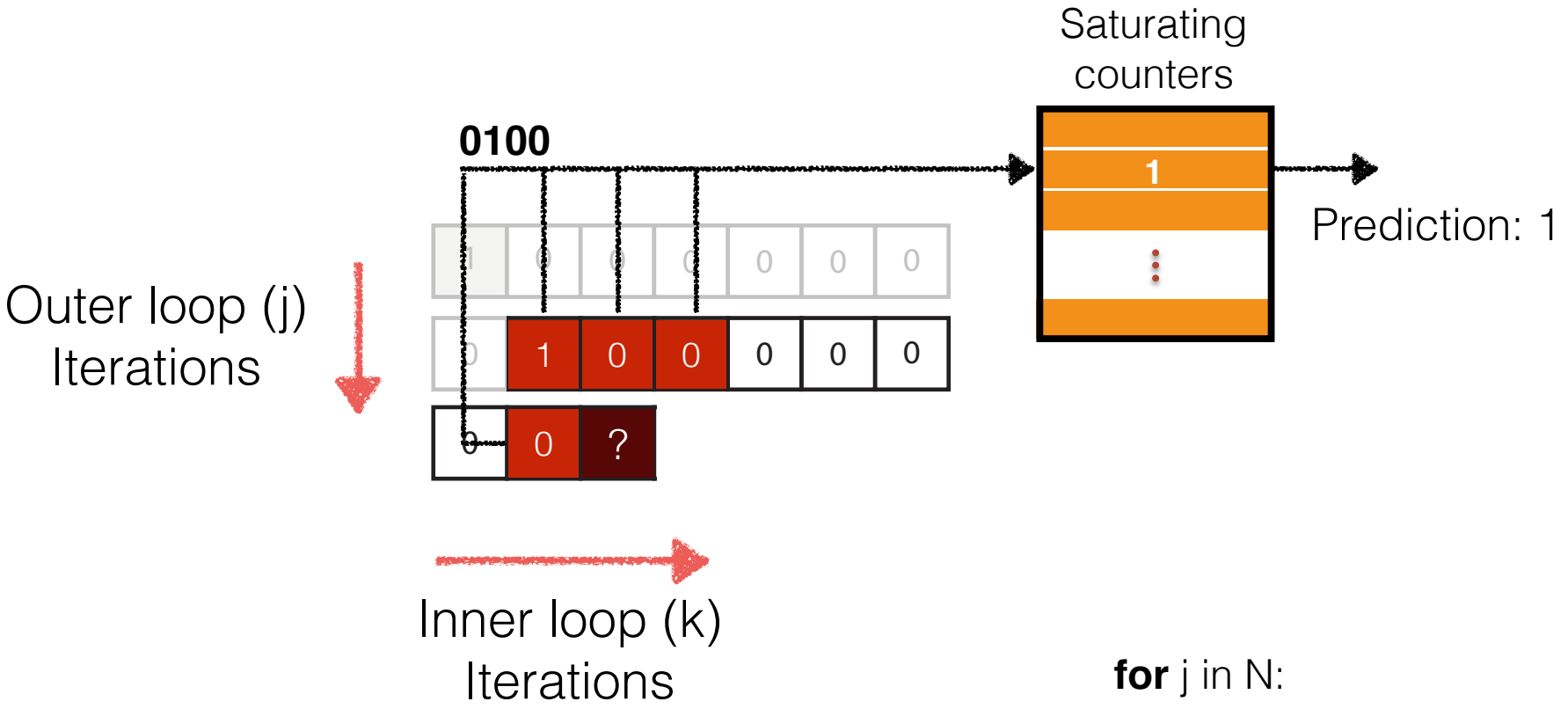

```
for j in N:
  for k in N:
    if  j != k:
      /*Calculation*/
```

Multidimensional histories



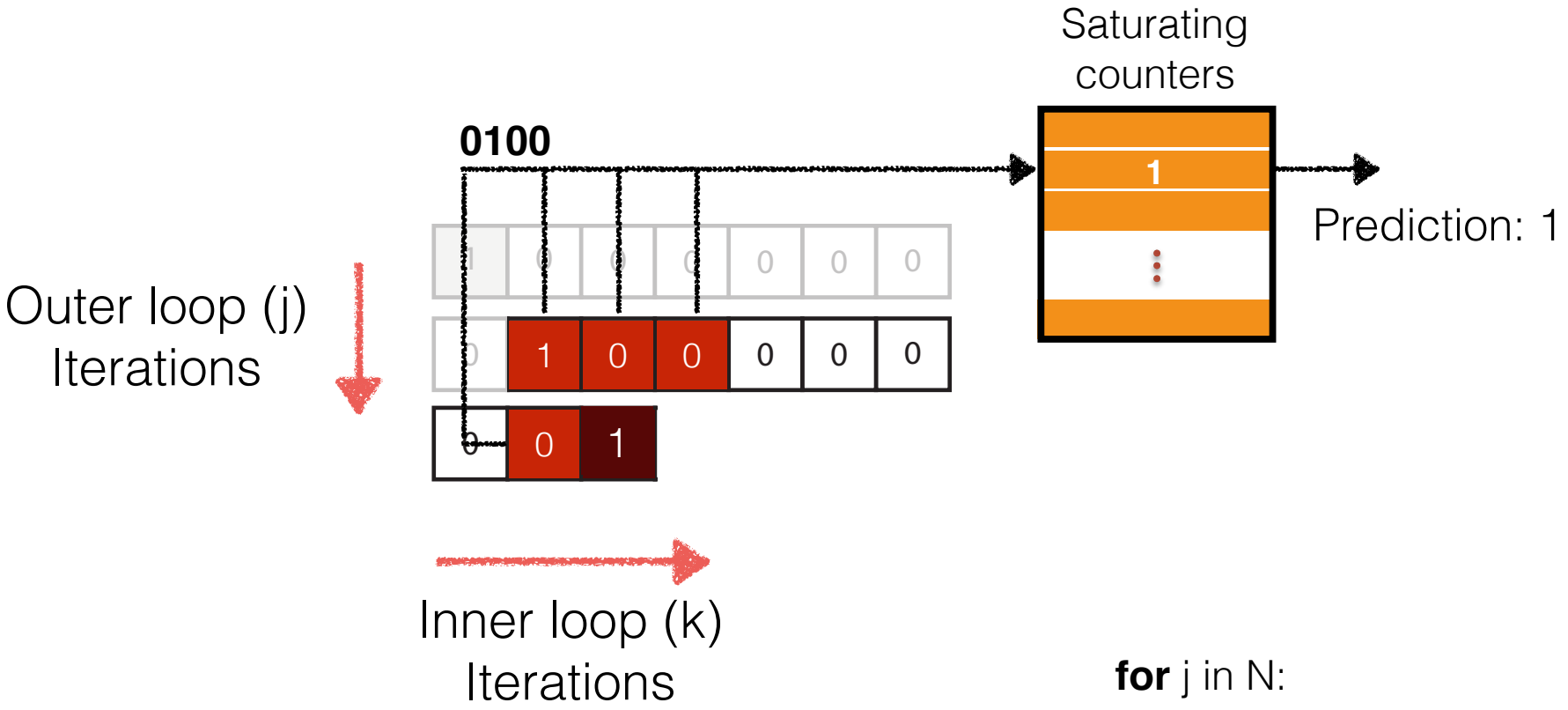

```
for j in N:
   for k in N:
      if  j != k:
         /*Calculation*/
```

Multidimensional histories



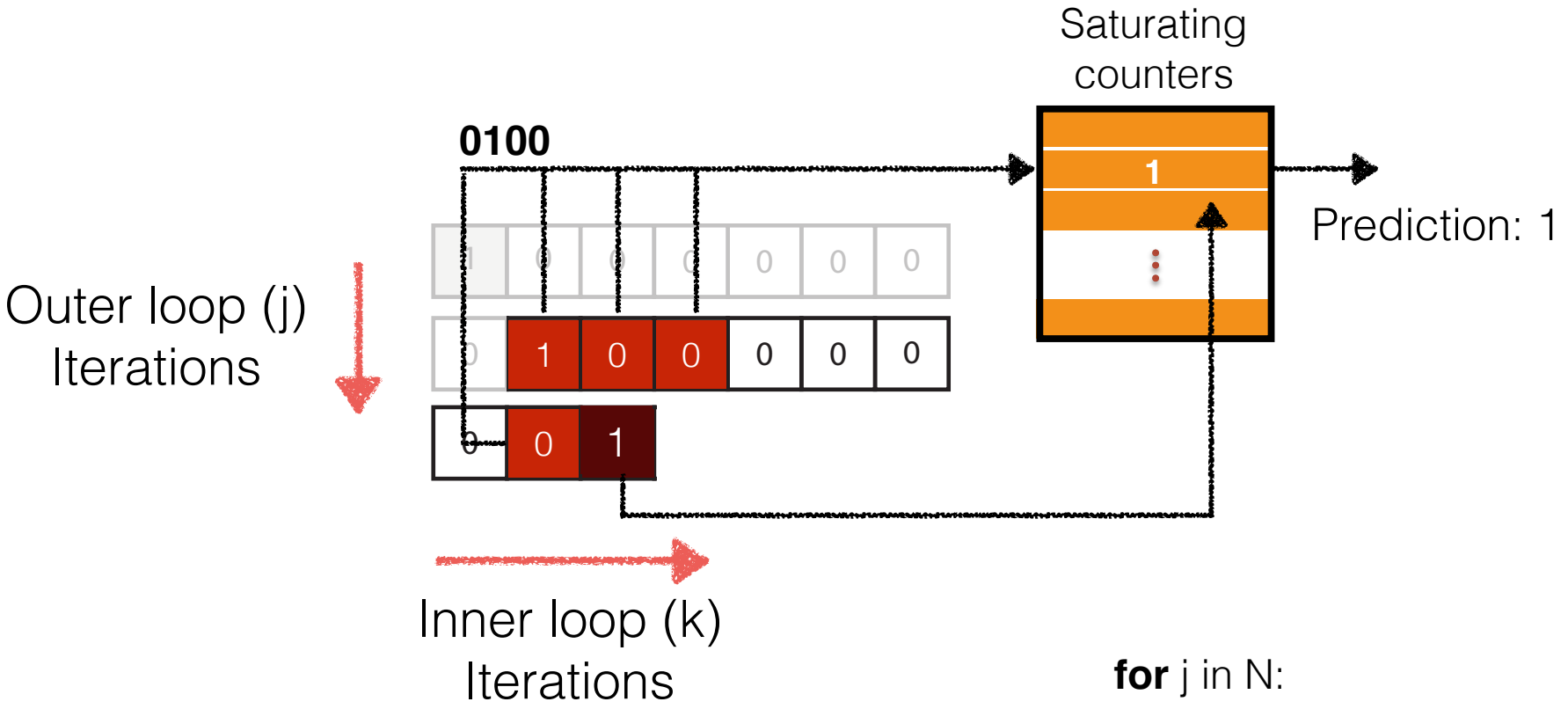

```
for j in N:
    for k in N:
        if  j != k:
            /*Calculation*/
```

Multidimensional histories




```
for j in N:
    for k in N:
        if  j != k:
            /*Calculation*/
```

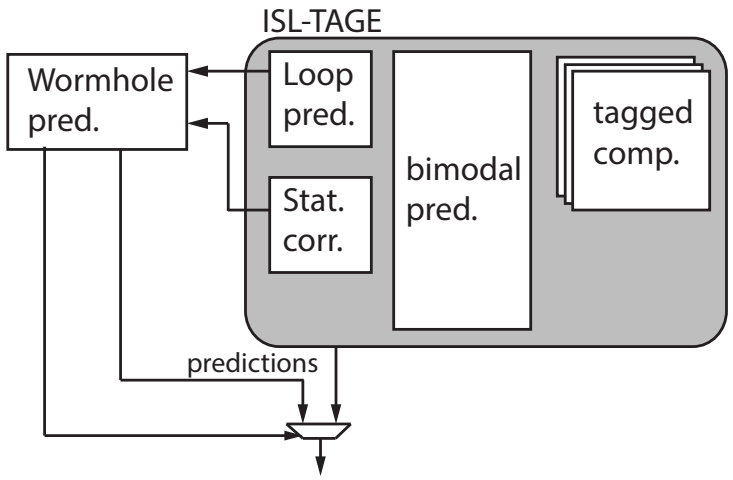
Multidimensional histories




```
for j in N:
    for k in N:
        if  j != k:
            /*Calculation*/
```

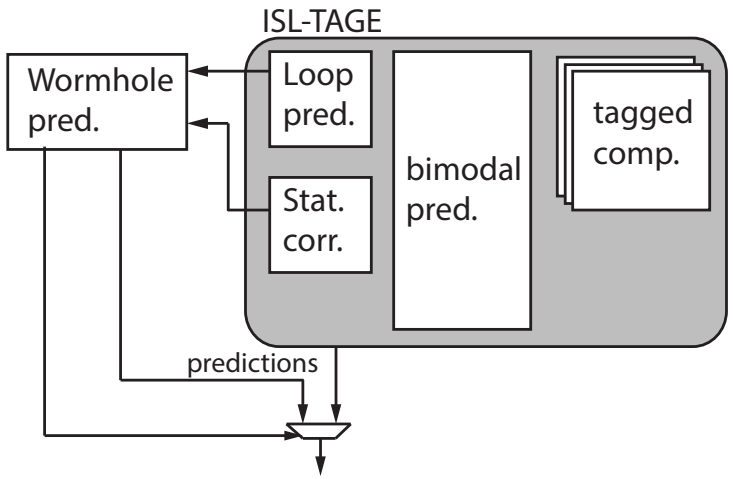
Wormhole predictor and ISL-TAGE

- On top of ISL-TAGE
  (Best performer from last CBP)
  - TAGE: bimodal + tagged
  - Statistical corrector
  - Loop predictor



Wormhole predictor and ISL-TAGE

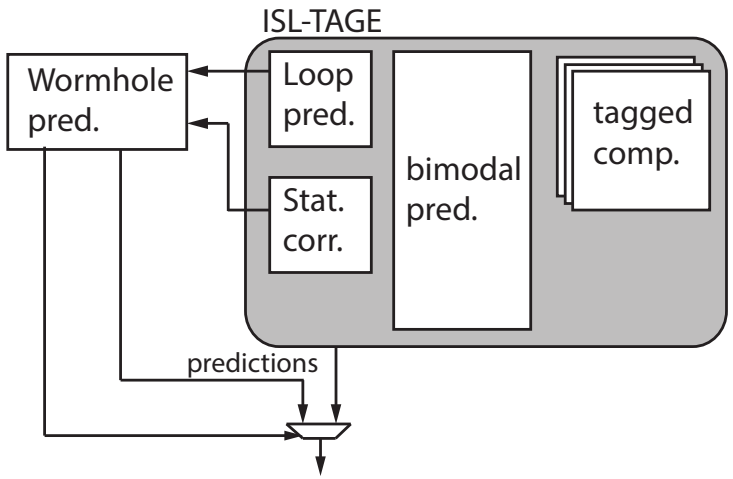
- On top of ISL-TAGE
  (Best performer from last CBP)
  - TAGE: bimodal + tagged
  - Statistical corrector
  - Loop predictor



Detect difficult branches

Wormhole predictor and ISL-TAGE

- On top of ISL-TAGE
  (Best performer from last CBP)
  - TAGE: bimodal + tagged
  - Statistical corrector
  - Loop predictor

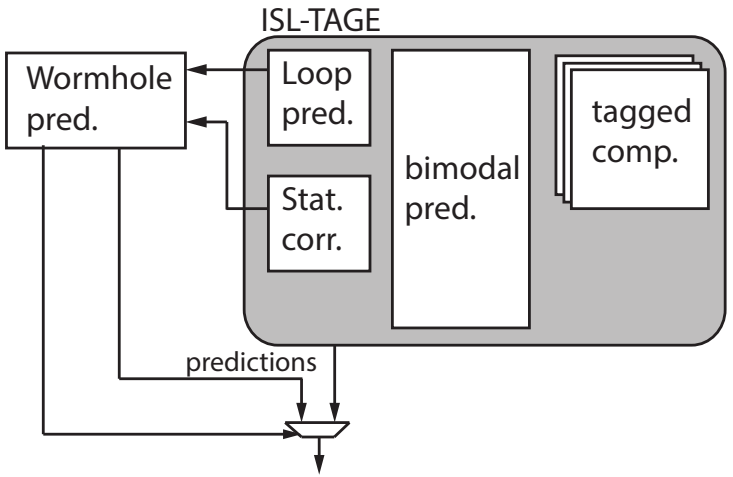


Detect difficult branches

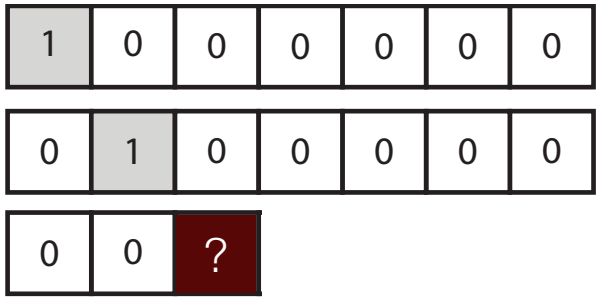
Loop dimensionality

ISL-TAGE

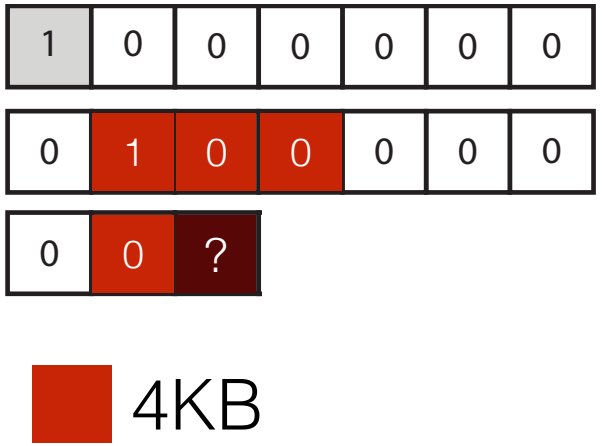
- **Difficult branches**
  - Statistical corrector calculates accuracy for TAGE
  - Wormhole allocates when low accuracy
- **Loop dimensionality**
  - Stored at *LP_iters*
- ***u* field (4KB)**
  - 3 bits
  - Updated when “as in the original TAGE”
  - Not reset: Incremented and decremented



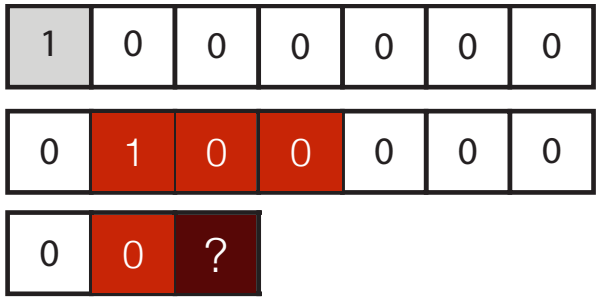
Wormhole predictor’s configurations



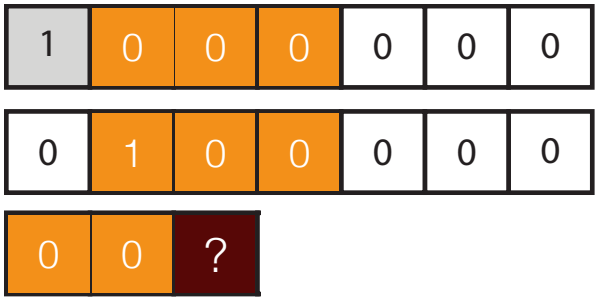
Wormhole predictor’s configurations



Wormhole predictor's configurations

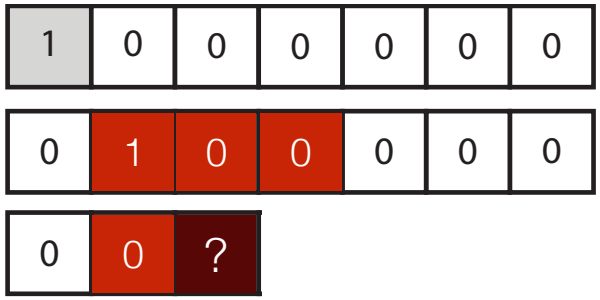


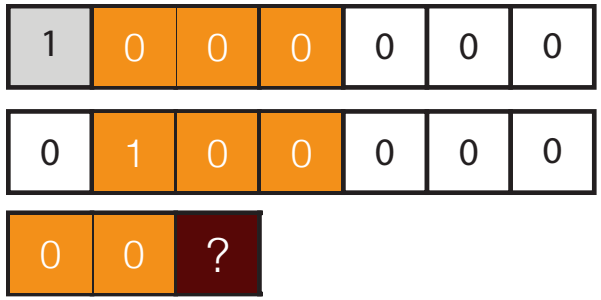
4KB



32KB & Unlim.

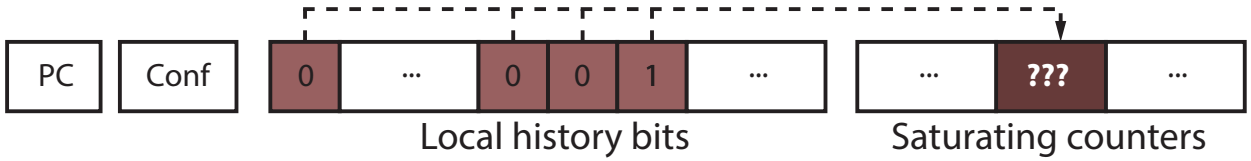
Wormhole predictor's configurations



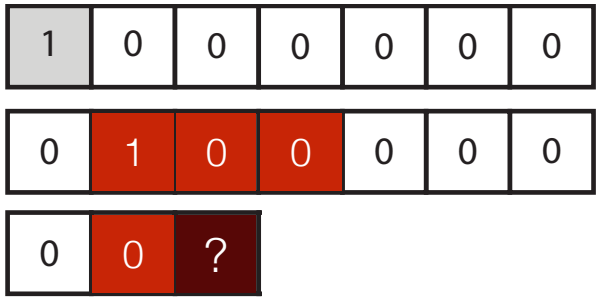


4KB

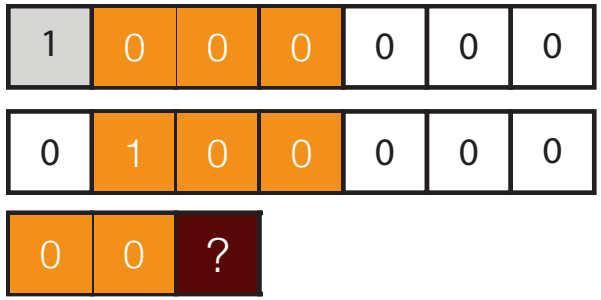
32KB & Unlim.



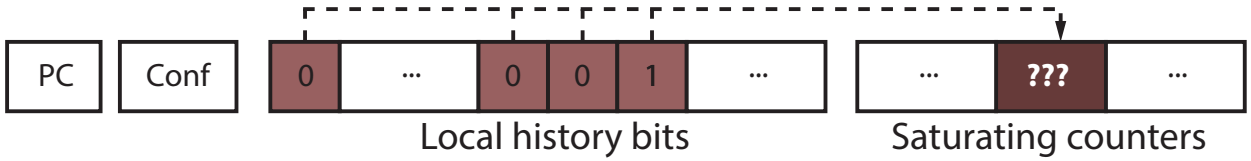
Wormhole predictor's configurations



4KB



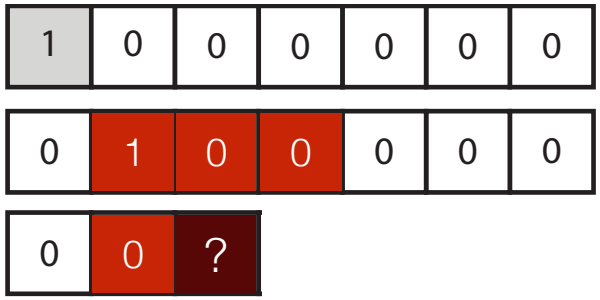
32KB & Unlim.

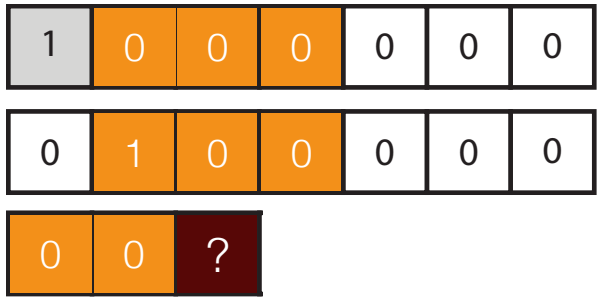


4KB:	18	4	101	16 x 5
32KB & Unlim.:	18	4	301	256 x 5

(bits)

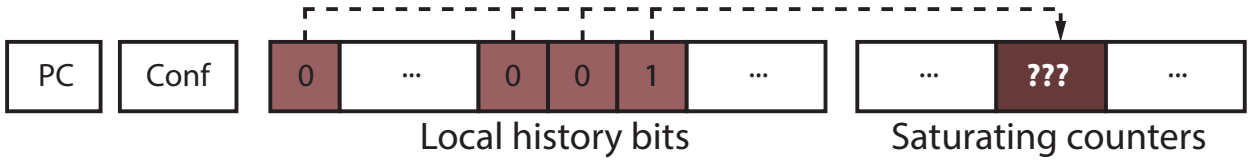
Wormhole predictor's configurations





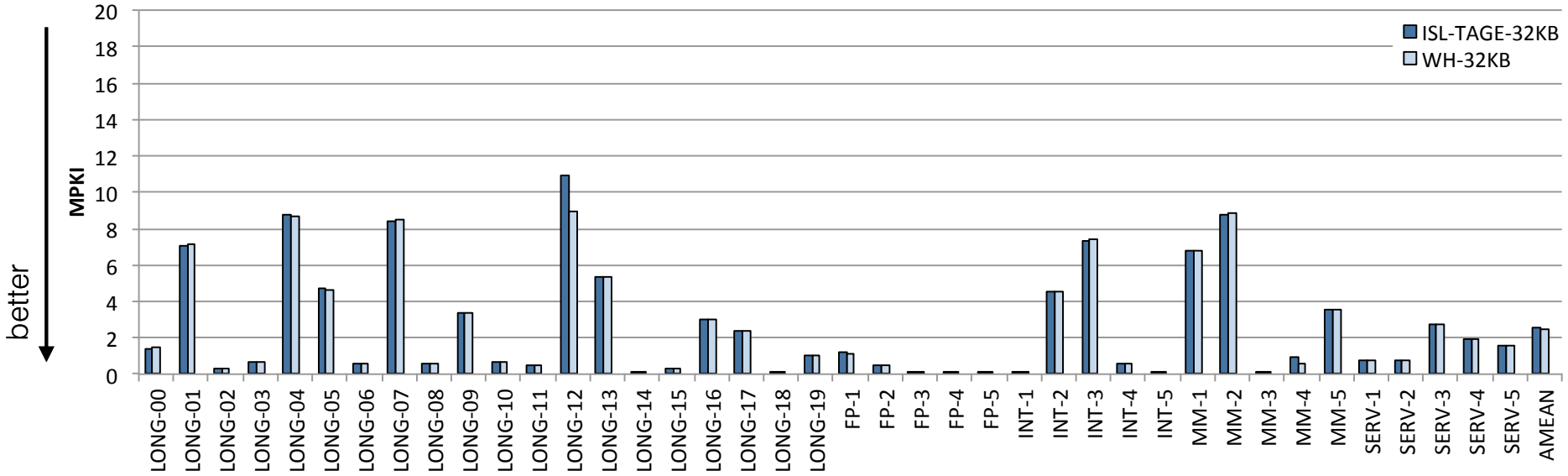
4KB

32KB & Unlim.

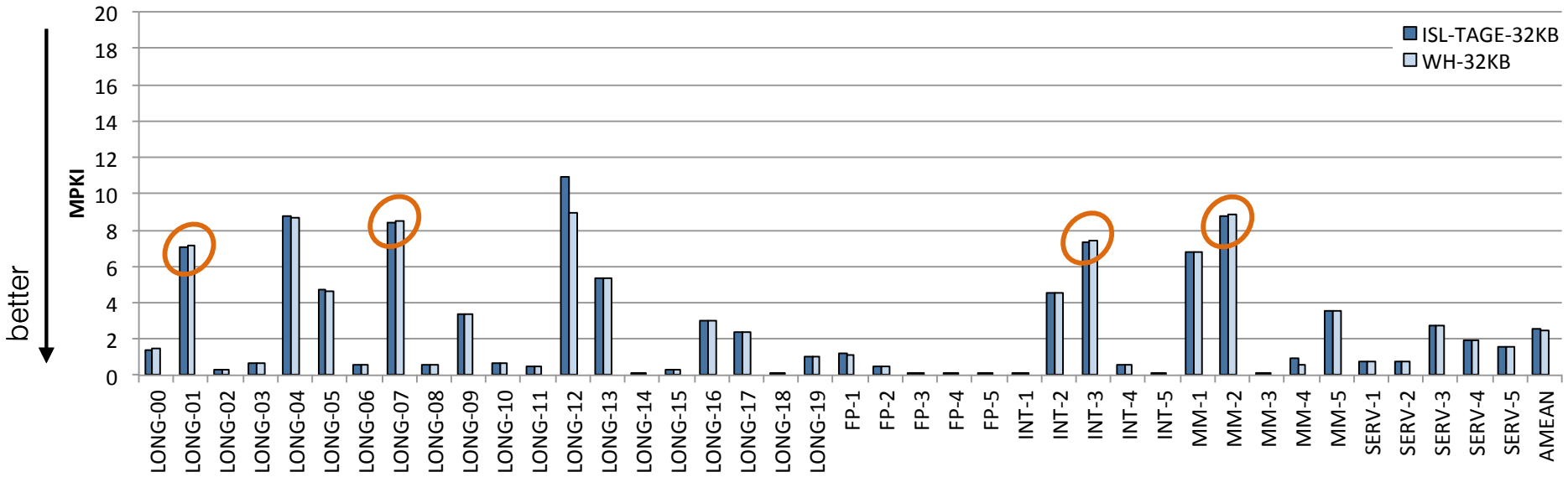


4KB:	18	4	101	16 x 5	= 197 (bits)
32KB & Unlim.:	18	4	301	256 x 5	= 1615

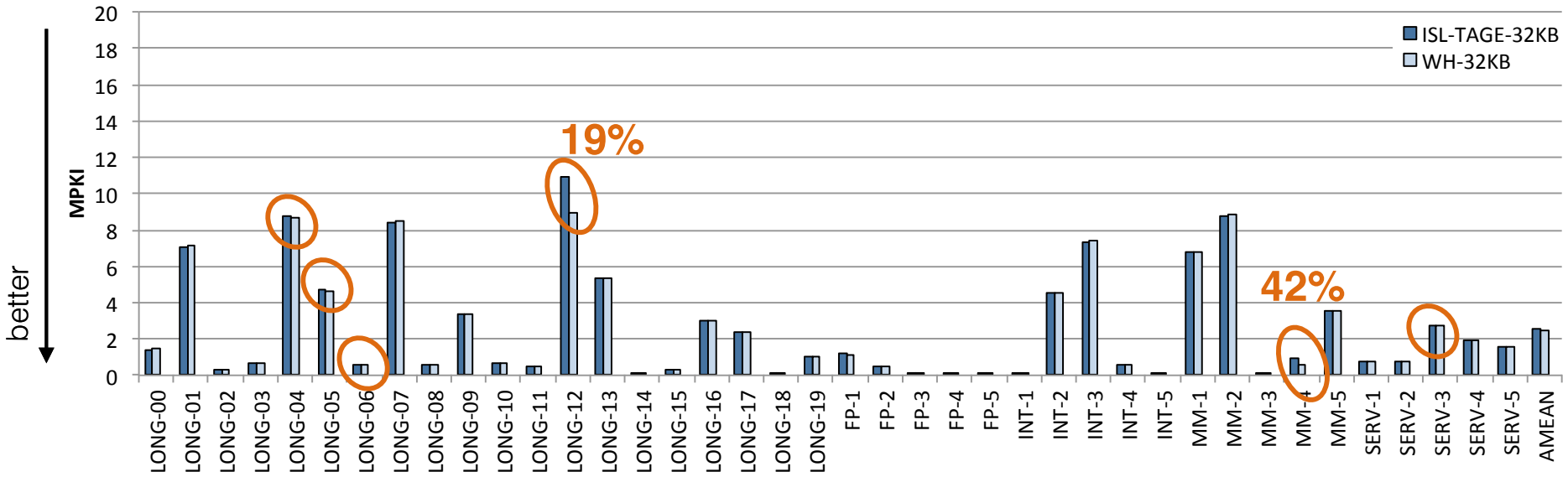
Results: 32KB



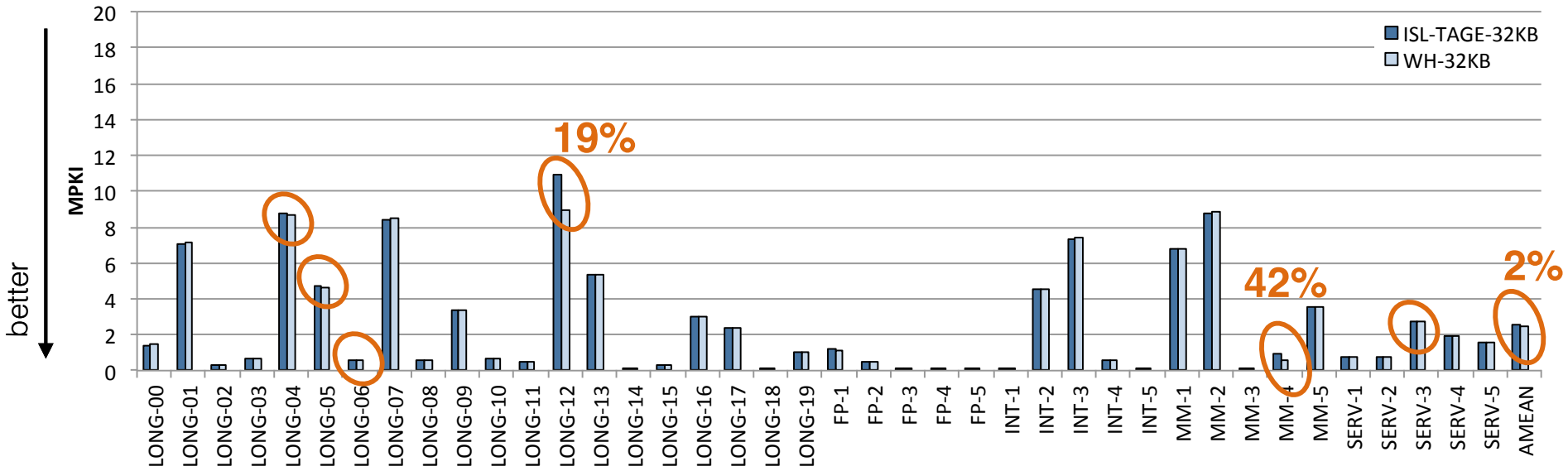
Results: 32KB



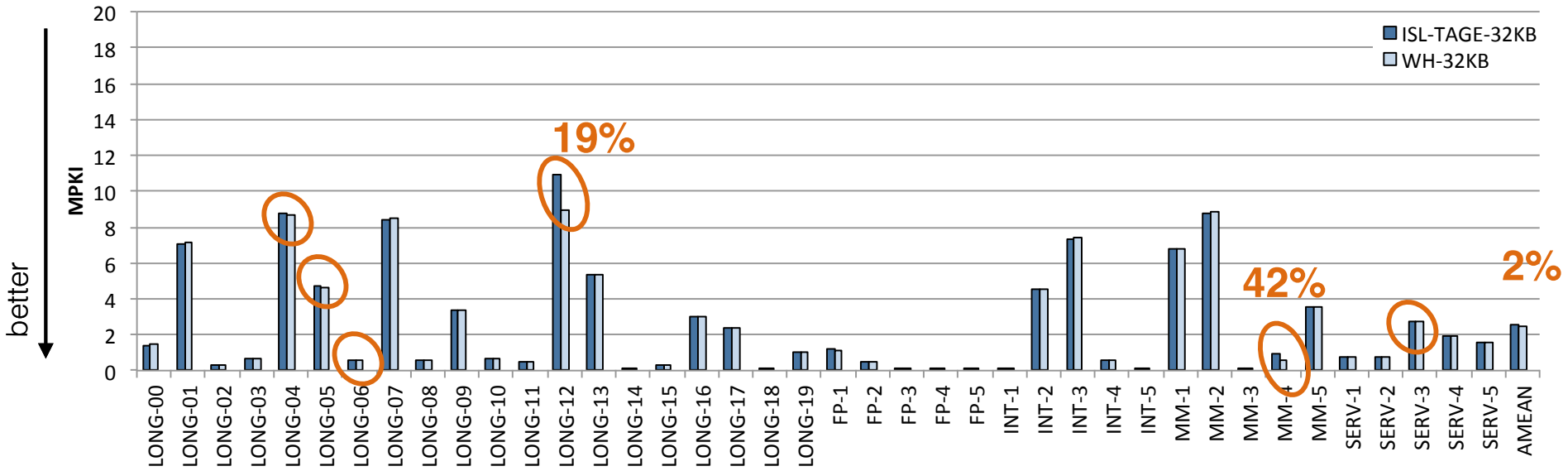
Results: 32KB



Results: 32KB



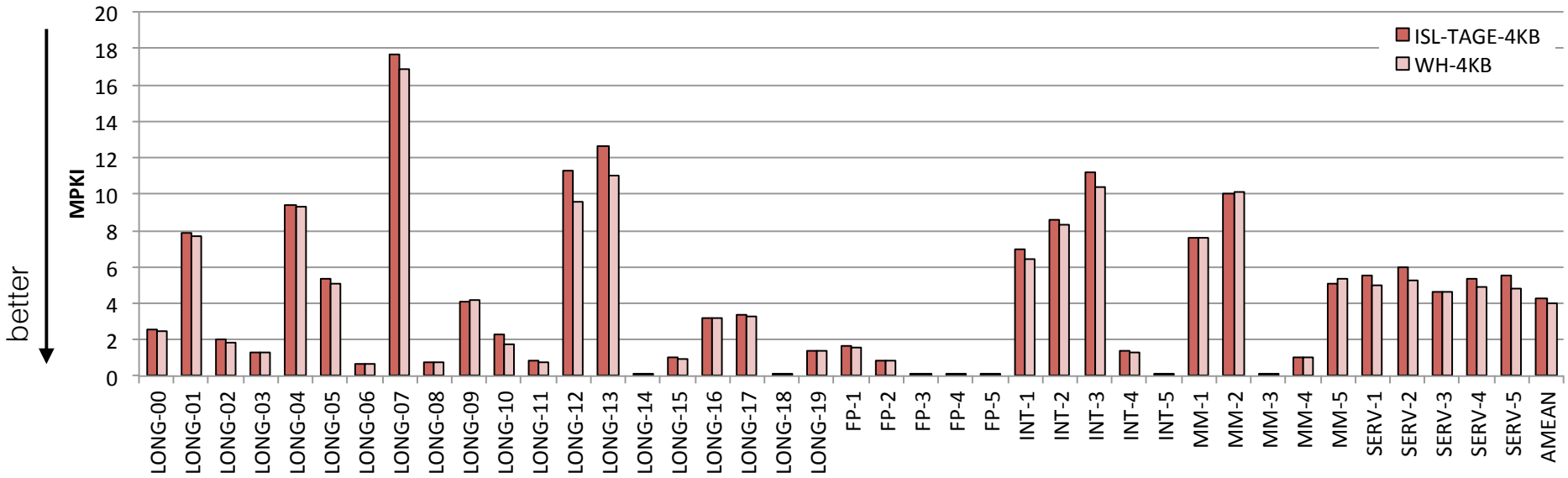
Results: 32KB



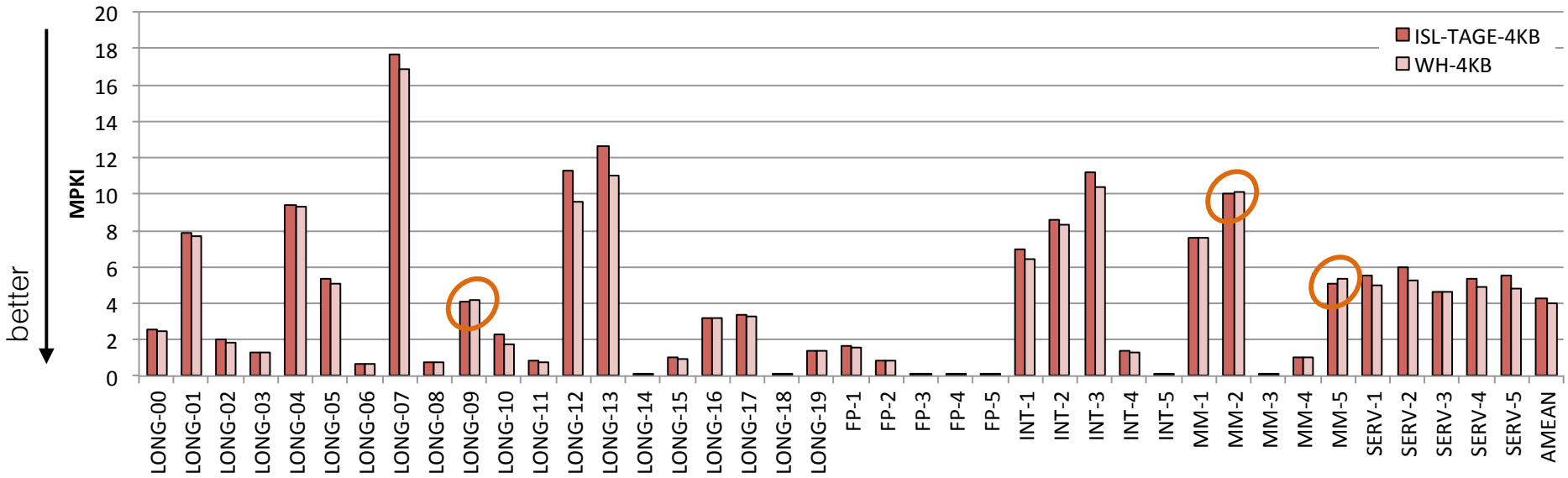
Big improvements on two traces

32KB: Avg. MPKI = 2.498

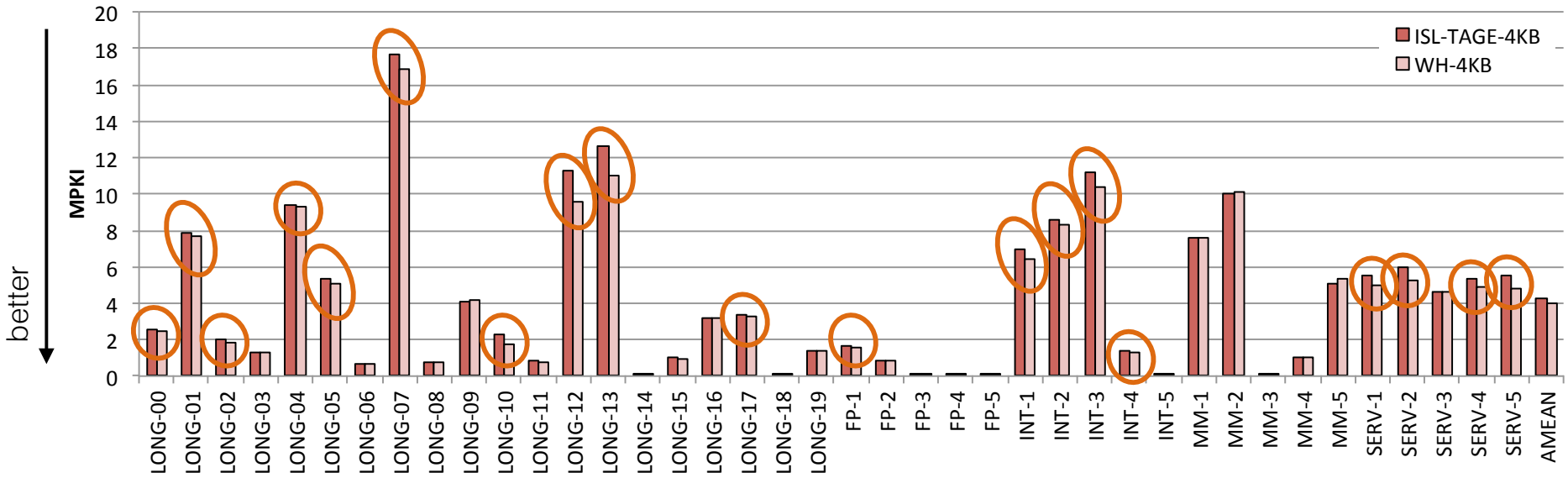
Results: 4KB


ISL-TAGE-4KB
WH-4KB
MPKI
better
20
18
16
14
12
10
8
6
4
2
0
LONG-00
LONG-01
LONG-02
LONG-03
LONG-04
LONG-05
LONG-06
LONG-07
LONG-08
LONG-09
LONG-10
LONG-11
LONG-12
LONG-13
LONG-14
LONG-15
LONG-16
LONG-17
LONG-18
LONG-19
FP-1
FP-2
FP-3
FP-4
FP-5
INT-1
INT-2
INT-3
INT-4
INT-5
MM-1
MM-2
MM-3
MM-4
MM-5
SERV-1
SERV-2
SERV-3
SERV-4
SERV-5
AMEAN

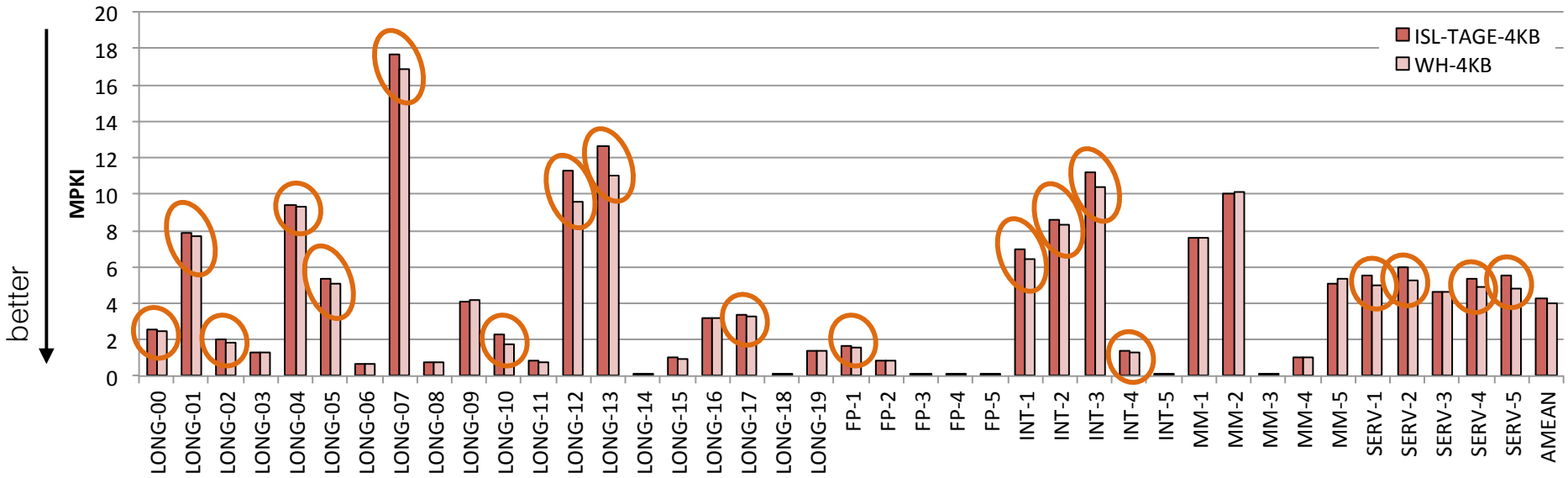
Results: 4KB


ISL-TAGE-4KB
WH-4KB
MPKI
better
20
18
16
14
12
10
8
6
4
2
0
LONG-00
LONG-01
LONG-02
LONG-03
LONG-04
LONG-05
LONG-06
LONG-07
LONG-08
LONG-09
LONG-10
LONG-11
LONG-12
LONG-13
LONG-14
LONG-15
LONG-16
LONG-17
LONG-18
LONG-19
FP-1
FP-2
FP-3
FP-4
FP-5
INT-1
INT-2
INT-3
INT-4
INT-5
MM-1
MM-2
MM-3
MM-4
MM-5
SERV-1
SERV-2
SERV-3
SERV-4
SERV-5
AMEAN

Results: 4KB


ISL-TAGE-4KB
WH-4KB
MPKI
better
20
18
16
14
12
10
8
6
4
2
0
LONG-00
LONG-01
LONG-02
LONG-03
LONG-04
LONG-05
LONG-06
LONG-07
LONG-08
LONG-09
LONG-10
LONG-11
LONG-12
LONG-13
LONG-14
LONG-15
LONG-16
LONG-17
LONG-18
LONG-19
FP-1
FP-2
FP-3
FP-4
FP-5
INT-1
INT-2
INT-3
INT-4
INT-5
MM-1
MM-2
MM-3
MM-4
MM-5
SERV-1
SERV-2
SERV-3
SERV-4
SERV-5
AMEAN

Results: 4KB



Better than ISL-TAGE in 20 out of 40 traces (4.6% on avg.)
4KB: Avg. MPKI = 3.995

Wormhole predictor's storage

	4KB	**32KB**
Statistical corrector	96	384
Loop predictor	376	376
TAGE predictor	3589	30476
Wormhole predictor	134	1413
Total size (bytes)	4195	32649

Conclusion

- **Wormhole predictor**:
  Side-predictor on top of ISL-TAGE

- Able to consider the branch history as multidimensional, capturing columns, diagonals and other patterns

- **3.995**, **2.498**, and **2.014** MPKI in the 4KB, 32KB, and unlimited competition tracks, respectively

Conclusion

- **Wormhole predictor**:
  Side-predictor on top of ISL-TAGE

- Able to consider the branch history as multidimensional, capturing columns, diagonals and other patterns

1	0	1	0	0
1	0	1	0	0
1	0	1	0	0

- **3.995**, **2.498**, and **2.014** MPKI in the 4KB, 32KB, and unlimited competition tracks, respectively

Wormhole branch prediction using multi-dimensional histories

Jorge Albericio, Joshua San Miguel,
Natalie Enright Jerger, and Andreas Moshovos

Questions?