Sensitivity of Cache Replacement Policies

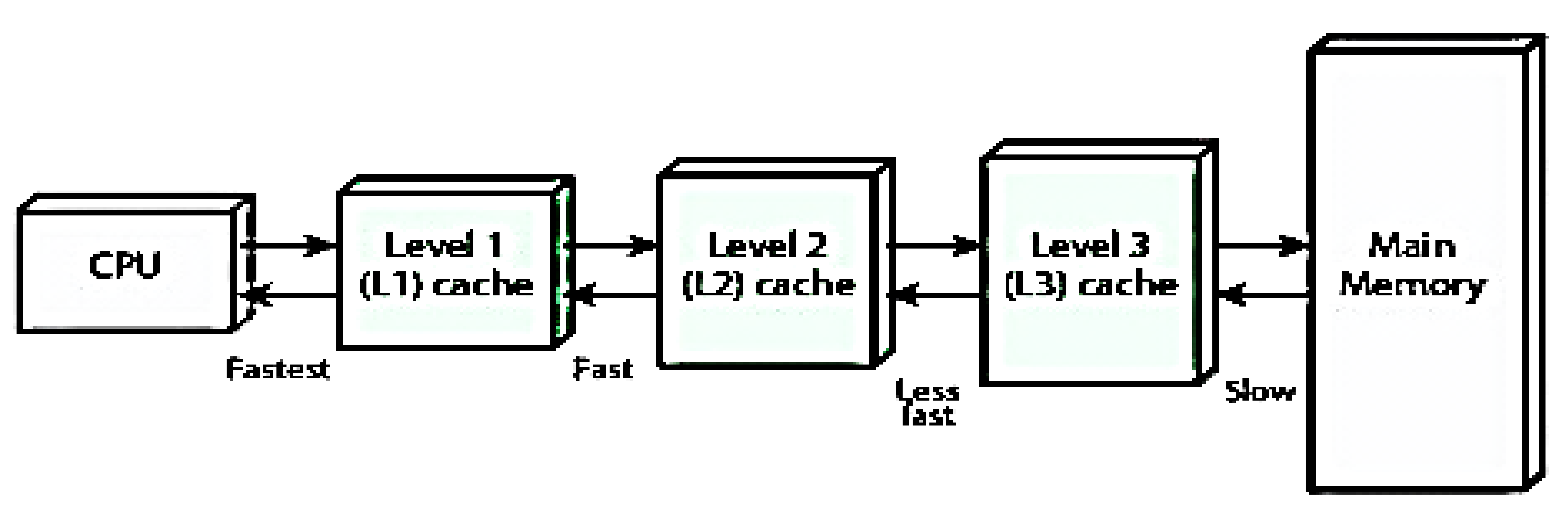
JAN REINEKE, DANIEL GRUND

Presented by Panchali Mukherjee
Saarland University

Robustness of Hardware and Software Systems

What is a cache?

- Memory unit.
- Conceals latency gap.
- Close to the CPU.
- Can access CPU faster.



(b) Three-level cache organization

Data found in cache: hit.
Data not found in cache: miss.

- Hit rate determines effectiveness.

What does Cache Miss result in?

- Very high cache miss penalties.
- Cache performance influences overall performance.

Worst Case Execution Time

The **maximum length of time** a task or set of tasks requires on a specific hardware platform.

Worst-case Execution Time Analysis:

Large variance gets introduced into the execution time due to:

- Cache misses
- Pipeline stalls, etc

Worst-case Execution Time Analysis:

WCET computation methods:

- Measurement-based timing Analysis
- Static Analysis.

Static Analysis:

- Uses abstract model.
- Computes invariant.
- Gives upper bound on WCET.

Static Analysis: Pros

- Gives good results for simple hardware.
- Efficient model.
- Accurate prediction.

Static Analysis: Cons

- Complex hardware: error prone and laborious.
- Inaccurate model.
- Over-pessimistic result.

Measurement Timing Analysis:

- Subset of real hardware states.
- Gives maximum of execution times measured.
- Underestimation of WCET.

Measurement Timing Analysis: Pro

- Gives good results for both simple and complex hardware.
- Precise estimate.
- Portable.

Measurement Timing Analysis: Con

- May suffer from over-pessimism.
- Not sound.

Worst-case Execution Time Analysis:

Objective:

Influence of initial hardware state on cache effectiveness.

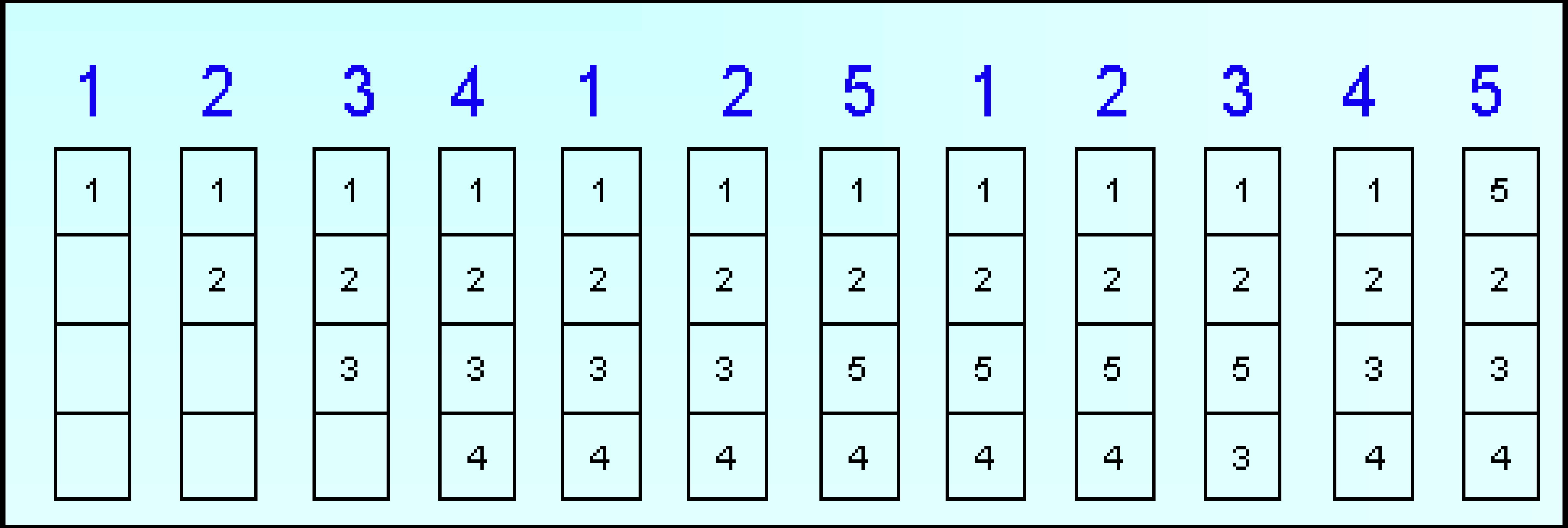
Associativity:

The size of a cache set is called the associativity k of the cache.

The index is used to find the set, and the tag helps find the block within the set.

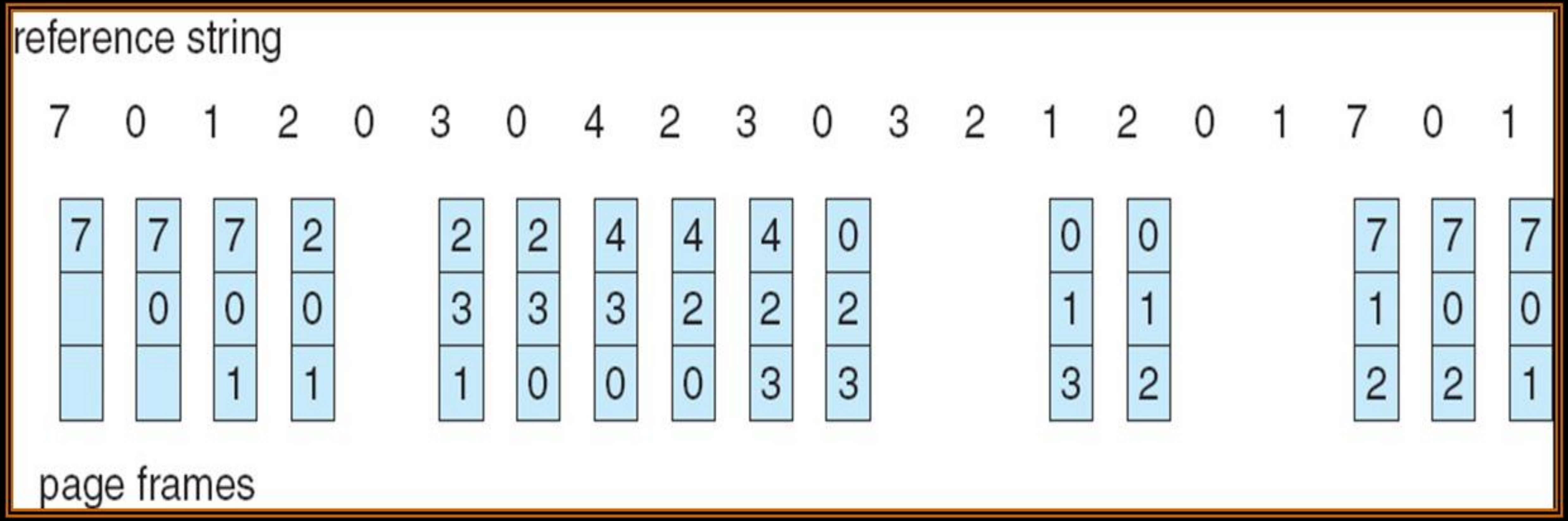
<u>Cache Replacement policies:</u>

LRU (least-recently-used): The bit that has not been used for the longest period of time is replaced.



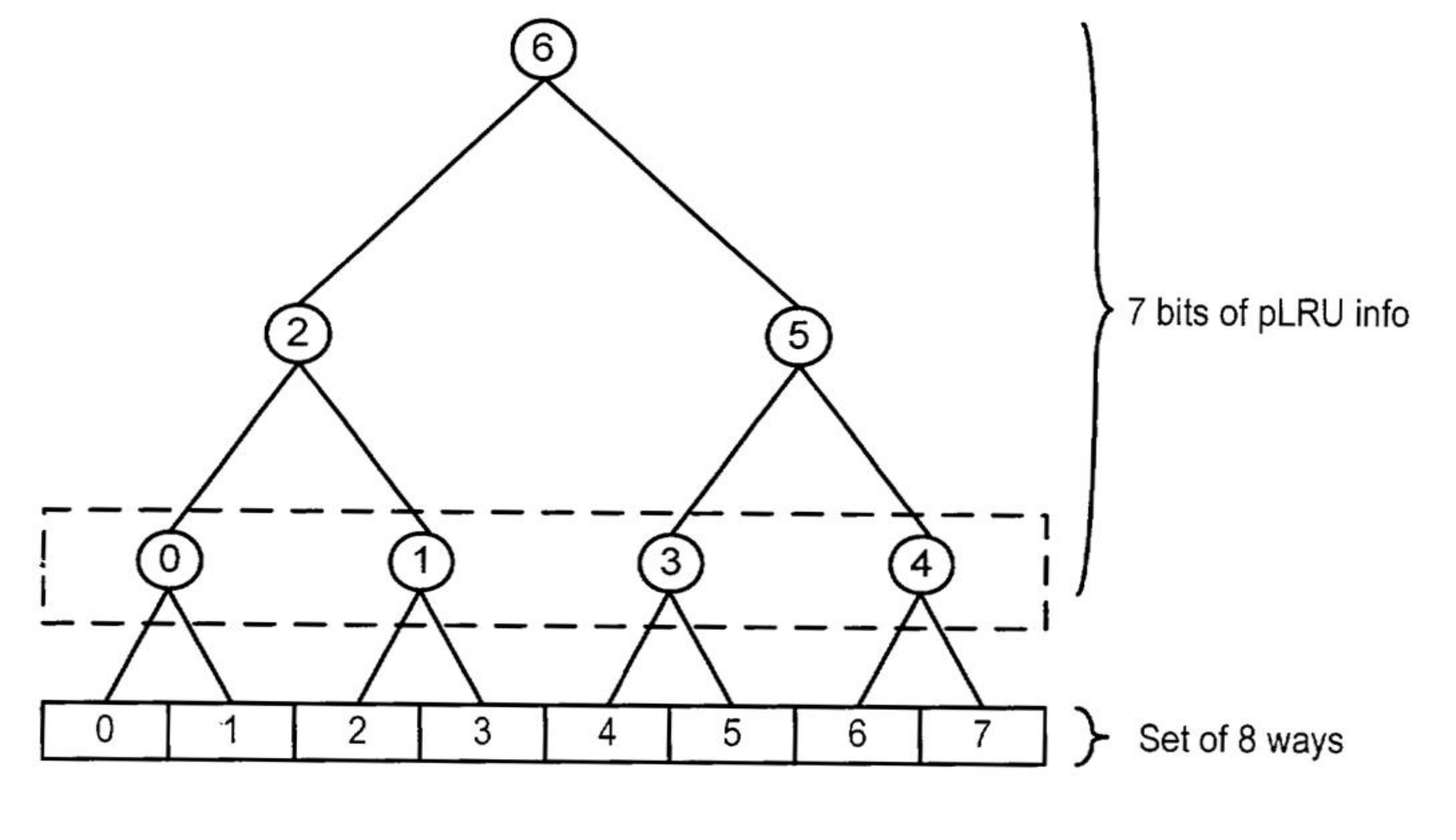
Cache Replacement policies:

FIFO (first-in-first-out): The bit that was the first to enter the cache is replaced.



Cache Replacement policies:

PLRU (Pseudo-LRU): Tree-based approximation of the LRU policy.



Cache Replacement policies:

MRU (most-recently-used):

- One recently-used bit per cache line.
- Cache line accessed—bit is set.
- Cache miss—first cache line without set bit selected,

  —block removed,

  —latest recently used bit set to 1.

  —all other bits reset to 0.

Sensitivity:

q, q' : cache states,

s: access sequence,

$m_P(q, s)$: number of misses,

$h_P(q, s)$: number of hits,

P: replacement policy used.

Sensitivity:

Miss-Sensitivity to State: A policy P is k-miss-sensitive with additive constant c, if

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

Sensitivity:

Hit-Sensitivity to State: A policy P is k-hit-sensitive with subtractive constant c, if

$$h_P(q, s) \geq k \cdot h_P(q', s) - c$$

Sensitivity:

Sensitive Ratio: The sensitive miss and hit ratios s_p^m and s_p^h of P are defined as:

$s_p^m = inf\ \{k \mid P \text{ is } k\text{-miss-sensitive}\}$ and

$s_p^h = sup\ \{k \mid P \text{ is } k\text{-hit-sensitive}\}$.

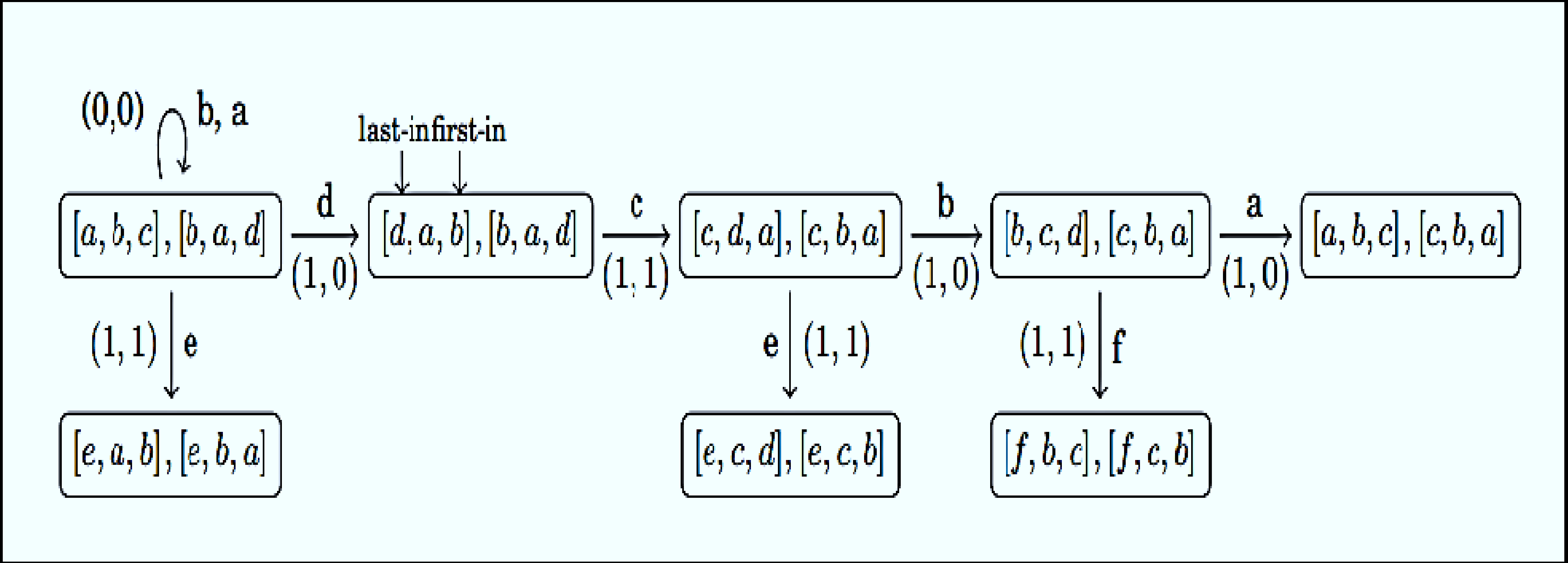
Compute Sensitive Ratios:

RELACS: Automatically computes sensitive ratios.

Compute Sensitive Ratios:

Transition system: **A system whose states are pairs of cache states, and whose transitions reflect the effect of a memory access on both of the cache states.**

Compute Sensitive Ratios:



Sensitive ratios depend on the number of misses (hits) on paths through the transition system.

Induced Transition System:

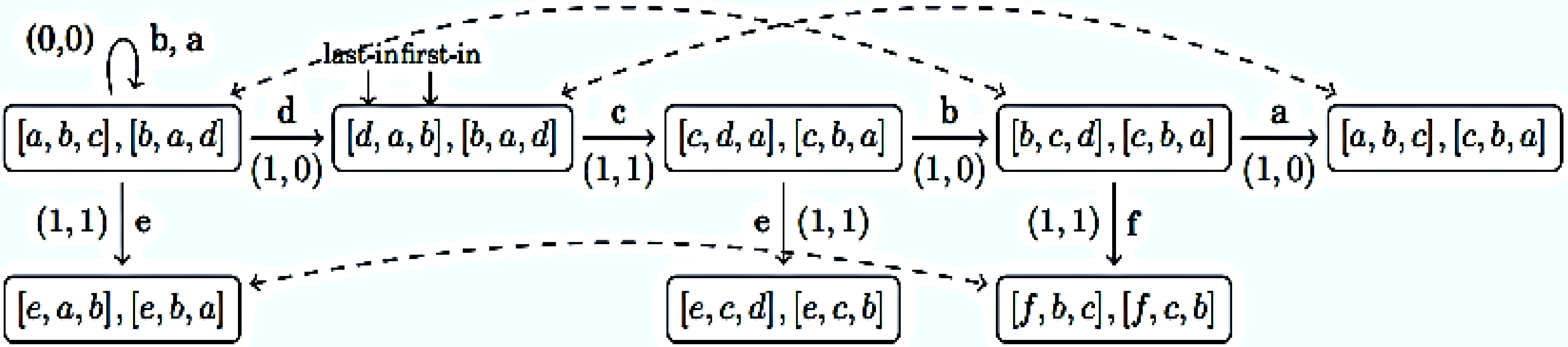
- Induced Transition System: A policy P induces a labeled transition system $T_P = (S_P , R_P)$, where:
- $S_P = \{(q, q') \mid q \in C_P , q' \in C_P\}$, ***are the states, which are pairs of cache states of policy P,***
- $R_P = \{((p, q), (m_p , m_q), (p' , q')) \mid (p, q) \in S_P , a \in B,$
- $(p' , q') = update_{P,P}((p, q), <a>)$
- $(m_p , m_q) = m_{P,P}((p, q), <a>)\}$

Induced Transition System:

If set of memory blocks infinite, the induced transition system is infinitely large.

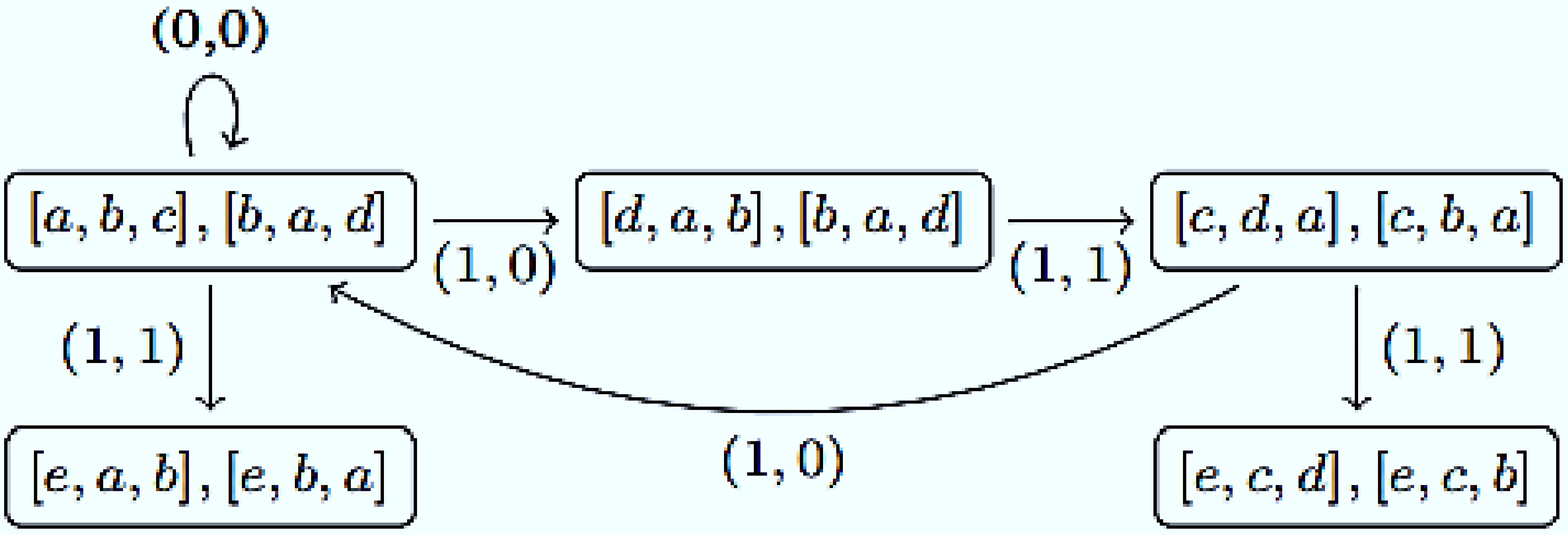
Solution: Finite Quotient Structure.

Merging Equivalent States:



(a) Dashed lines connect equivalent states according to the equivalence relation $\approx$.

Merging Equivalent States:

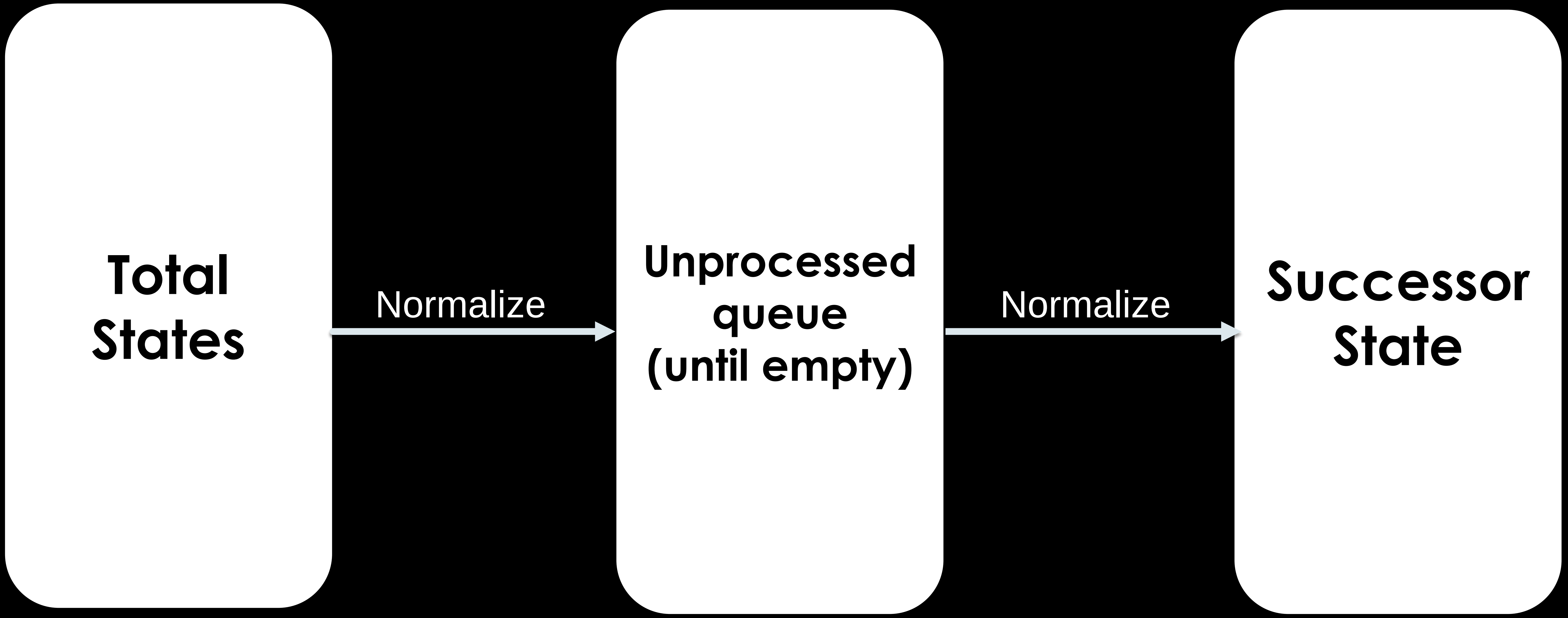


(b) Quotient structure.

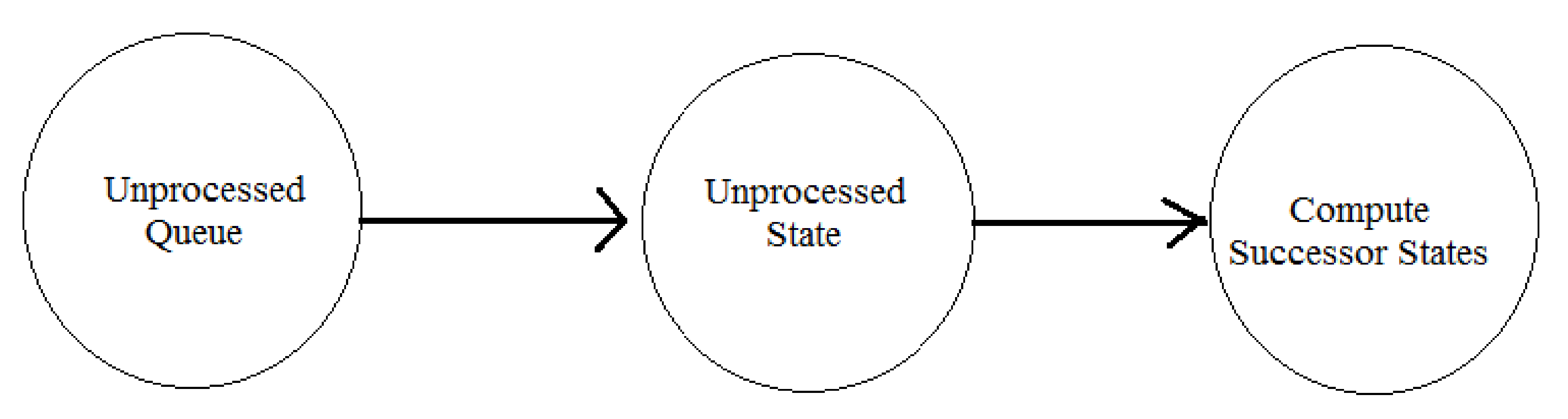
Quotient Transition System:

- Constructing a Quotient Transition System consists of two steps:
- 1. The computation of S_P
- 2. The computation of R_P .

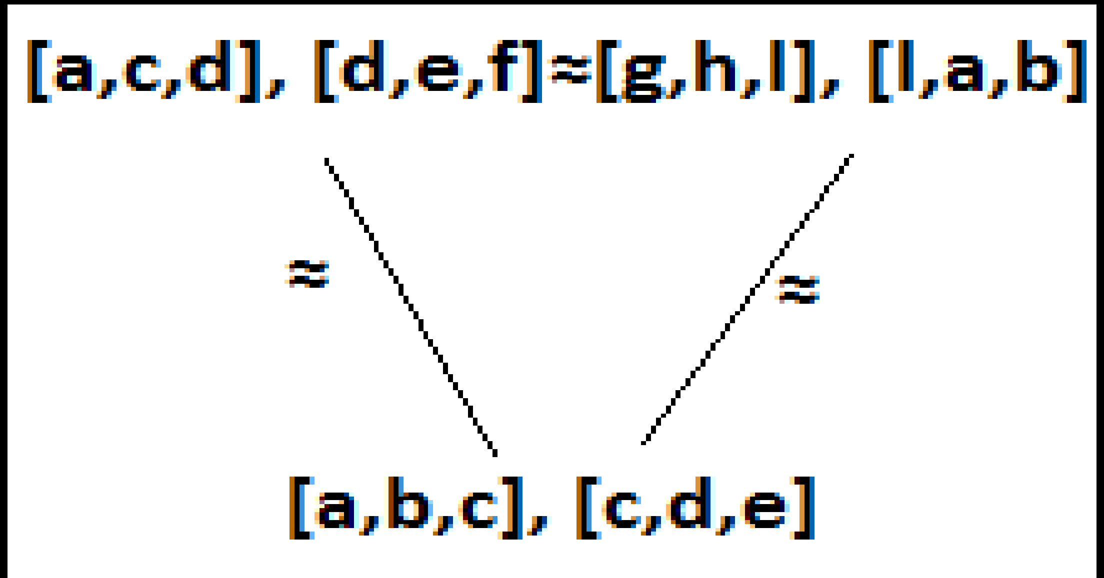
Quotient Transition System:



Quotient Transition System:



Computation of S_P



Normalize: Unique representative in the equivalence relation for pairs of states.

Quotient Transition System:

- Computation of R_P:

- NORMALIZE(update$_P$(p,<a>); update$_P$(q,<a>)) is equal for all a.

- Computing successors under the finite number of accesses.

Results:

- Sensitivity results for LRU, FIFO, PLRU, and MRU at associativity ranging from 2 to 8 is obtained.

- The data obtained are precise sensitive ratios.

Results:

Policies/Associativity	2	3	4	5	6	7	8
LRU	1,2	1,3	1,4	1,5	1,6	1,7	1,8
FIFO	2,2	3,3	4,4	5,5	6,6	7,7	8,8
PLRU	1,2	-	∞	-	-	∞	-
MRU	1,2	3,4	5,6	7,8	MEM	MEM	MEM

Miss-Sensitivity ratio, k, and additive constant, c, for FIFO, PLRU, and LRU.

Note: MEM indicates the algorithm ran out of memory on a 2gb machine.

Results:

Policies/Associativity	2	3	4	5	6	7	8
LRU	1,2	1,3	1,4	1,5	1,6	1,7	1,8
FIFO	0,0	0,0	0,0	0,0	0,0	0,0	0,0
PLRU	1,2	-	1/3, 5/3	-	-	∞	1/11, 19/11
MRU	1,2	0,0	0,0	0,0	MEM	MEM	MEM

Hit-Sensitivity ratio k, and subtractive constant c, for FIFO, PLRU, and LRU.

Note: MEM indicates the algorithm ran out of memory on a 2gb machine.

Results:

LRU is best replacement policy, most robust.

Open Question:

When the access sequence is restricted computing precise sensitive ratios become difficult.

Computing a quotient transition system, as done in this paper becomes improbable.

Common Ground:

ROBUSTNESS


Thank you!