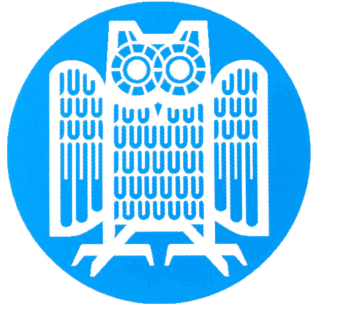




CAMA: Cache-Aware Memory Allocation for WCET Analysis



Jörg Herter Jan Reineke Reinhard Wilhelm
Department of Computer Science, Saarland University

WCET Analysis & Standard Malloc

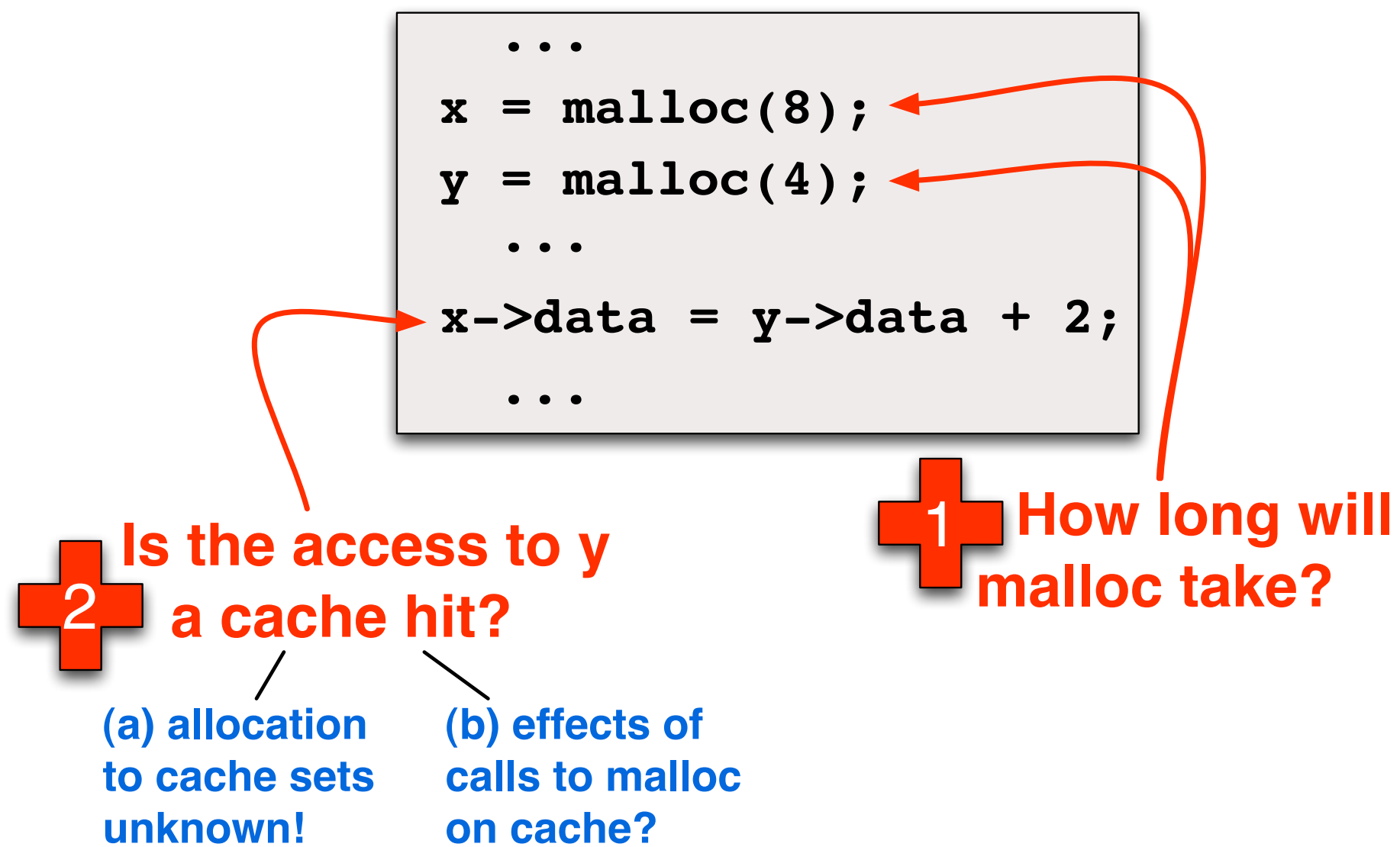
Current Situation

WCET analysis in the presence of static memory allocation only

Desired Situation

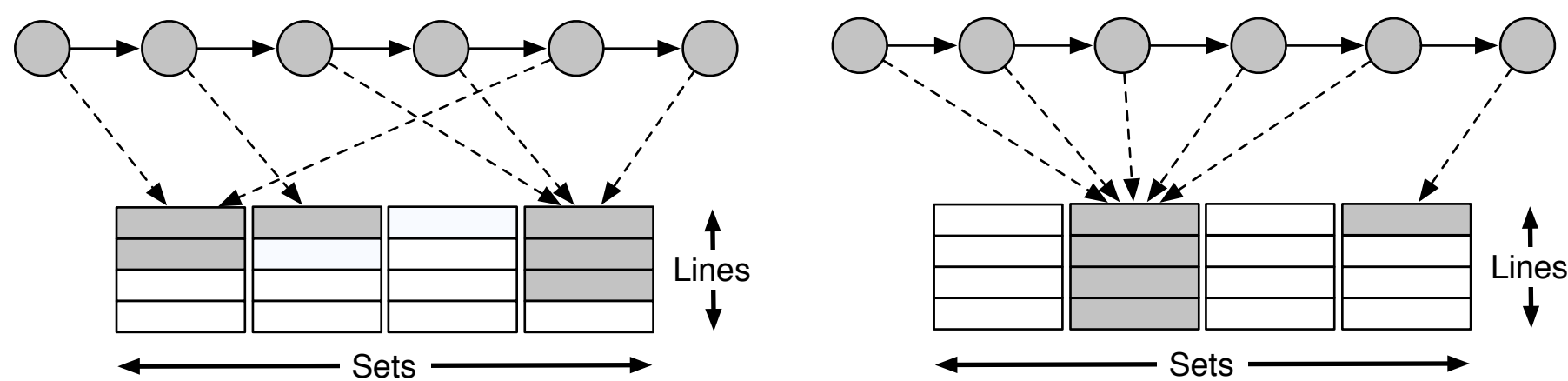
WCET analysis in the presence of dynamic memory allocation

Challenges



Examples

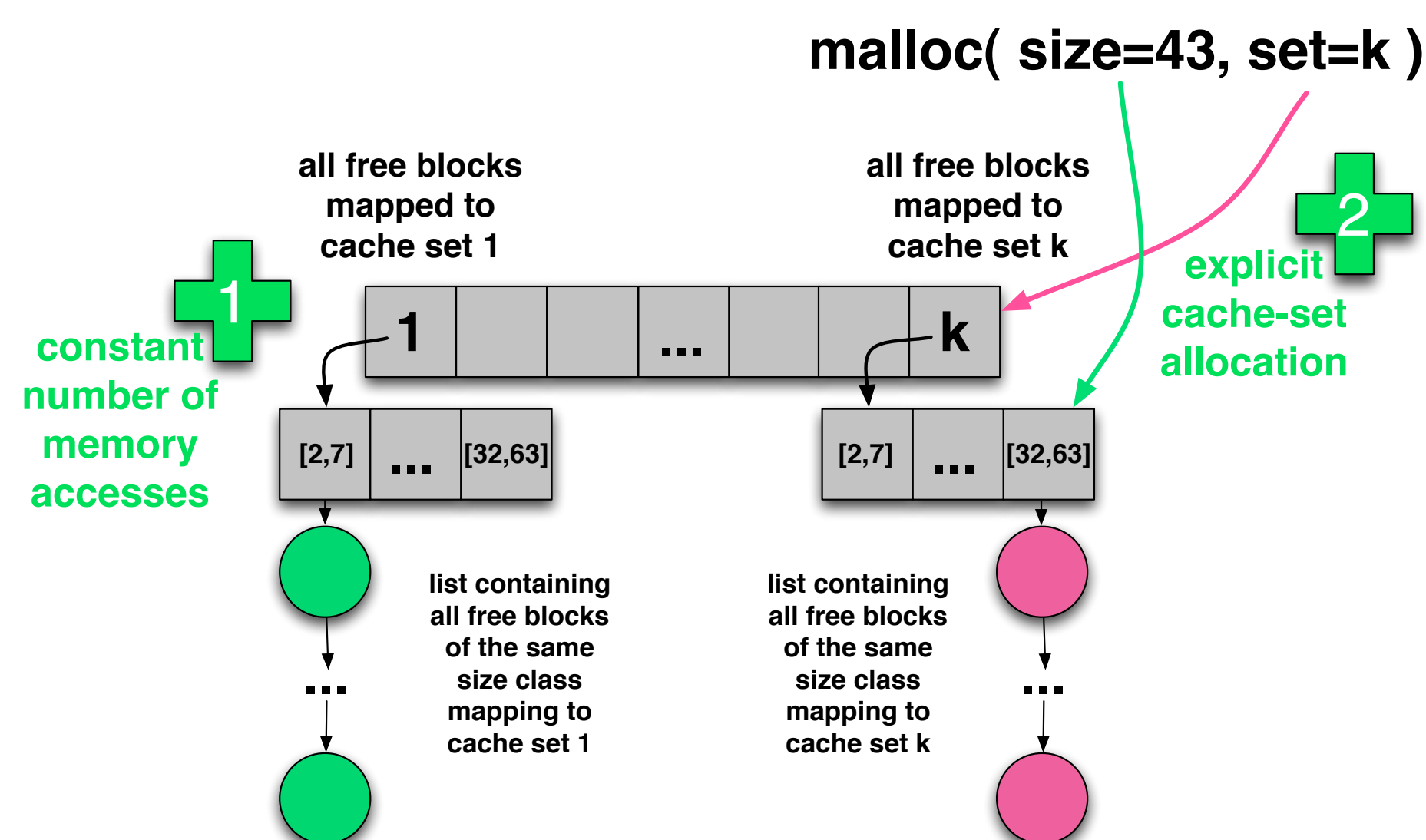
2 possible mappings of elements of a list structure to cache sets, assuming a 4-way-set-associative cache.



Cache-Aware Memory Allocation

Approach

1. Allocate from segregated lists \Rightarrow constant number of memory accesses, \approx constant execution time
2. Add new parameter to **malloc**: the cache set to allocate to



Contact information:

<http://rw4.cs.uni-sb.de/>
{jherter|reineke|wilhelm}@cs.uni-sb.de

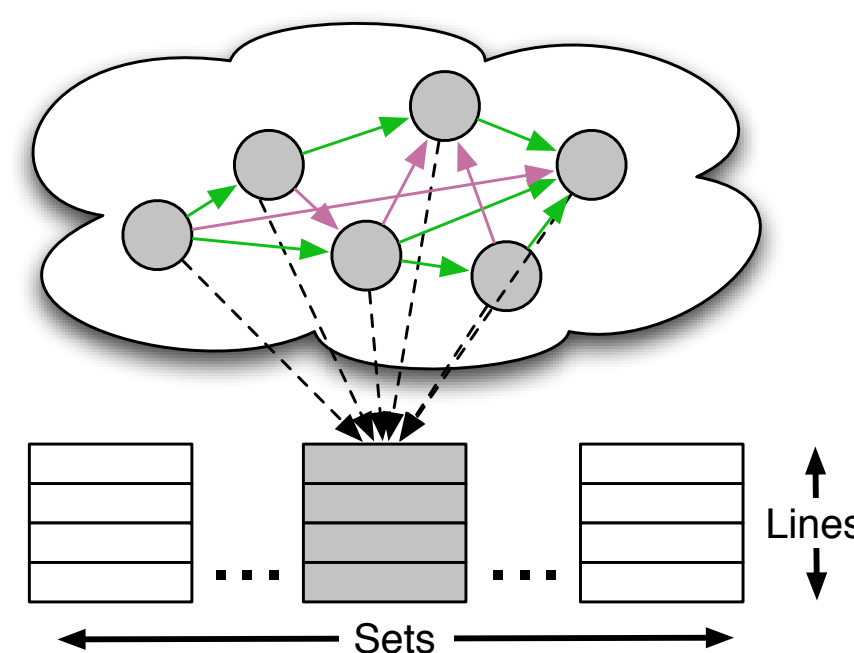
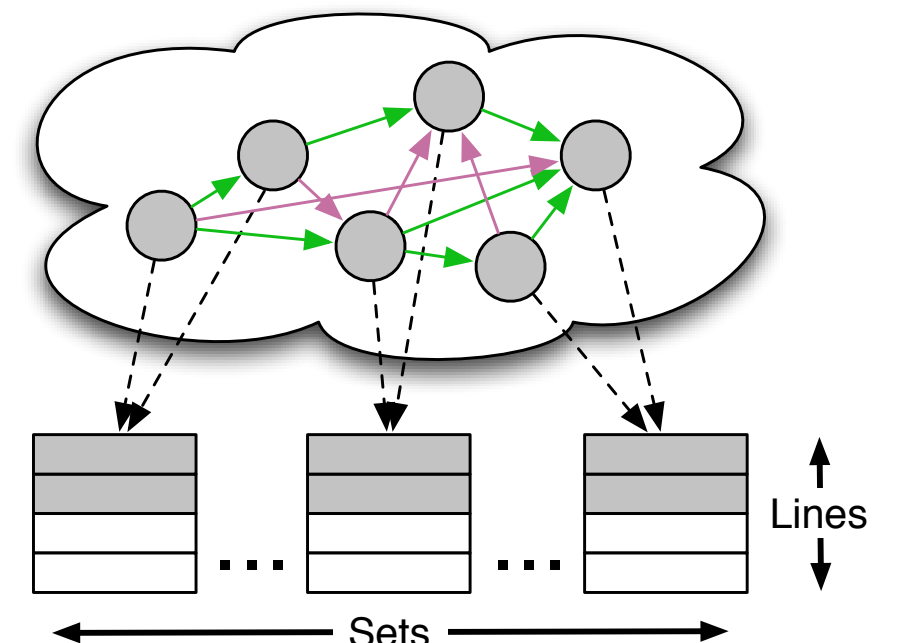
WCET Analysis & CAMA

How to Use Cache-Aware Memory Allocation

Control mapping of linked data structures to cache sets, to

- ... guarantee cache hits for those data structures,
- ... preserve information about other structures in the cache.

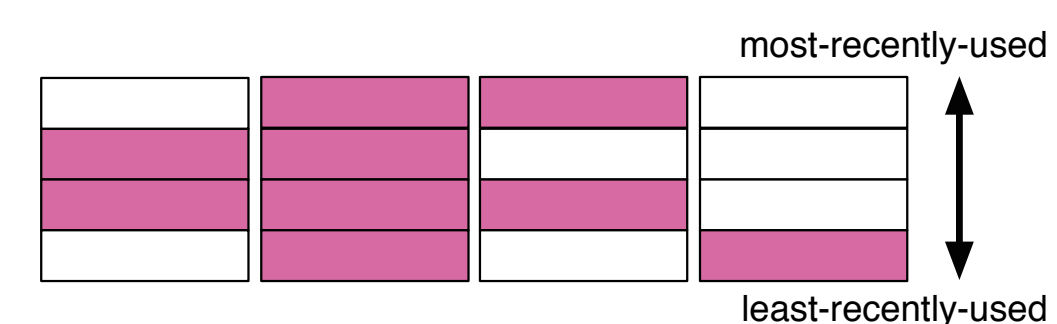
Mapping Scheme 1:
evenly distributing elements of a structure over all cache sets.



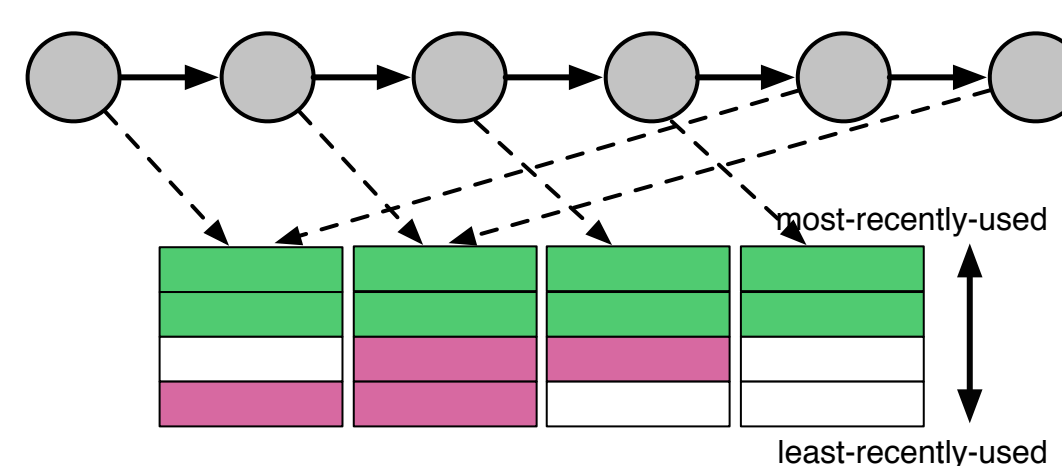
Mapping Scheme 2:
mapping all elements of a structure to a single cache set. All information about the cache – except about a single cache set – stays valid.

Example

Assume a 4-way-set-associative cache with 4 cache sets and a linked-list structure consisting of 6 elements.

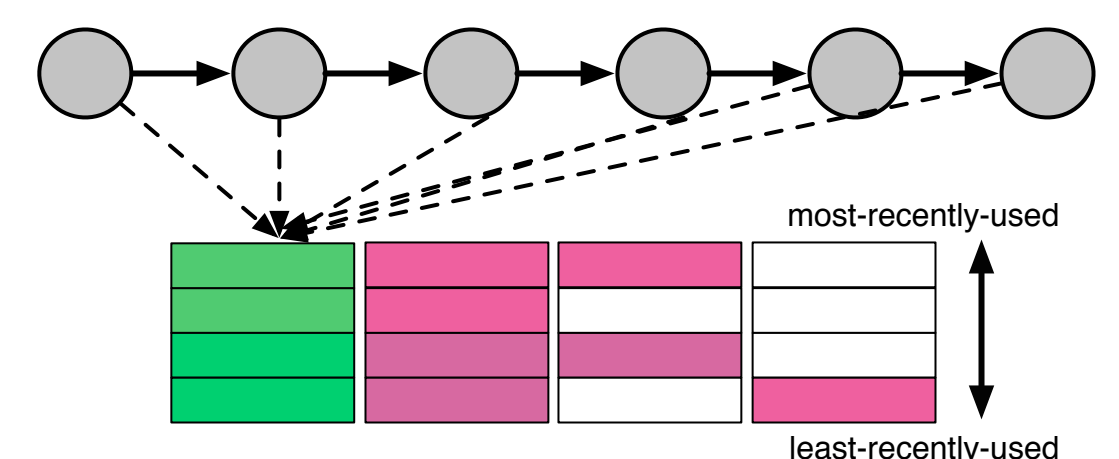


Initial cache state before list traversal.



Scheme 1 preserves information about half of the cache as at most 2 cache lines per cache set are affected.

Scheme 2 loses information about a whole cache set, but in turn preserves all information about the other sets.



Reference

Jörg Herter, Jan Reineke, and Reinhard Wilhelm:

CAMA: Cache-Aware Memory Allocation for WCET Analysis.

In *WIP-Proceedings of the Euromicro Conference on Real-Time Systems 2008.*

