Optimizing the JVM Object Model Using Object Splitting

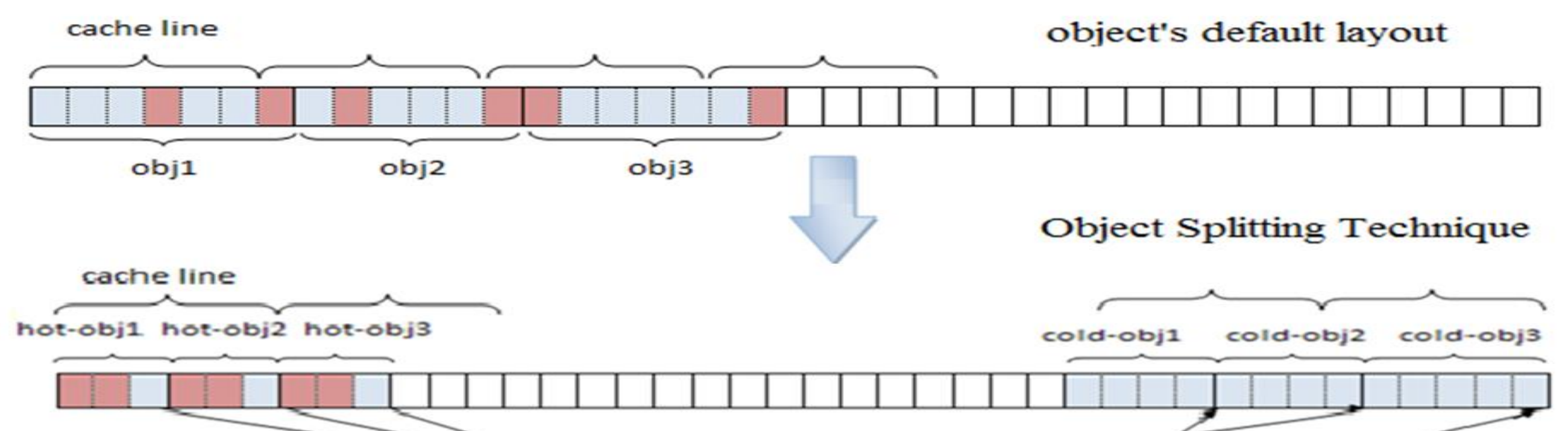
Taees Eimouri, Dr. Kenneth B.Kent
University of New Brunswick, Faculty of Computer Science
Aleksandar Micic
IBM Canada
teimouri@unb.ca, ken@unb.ca, aleksandar_micic@ca.ibm.com

ABSTRACT

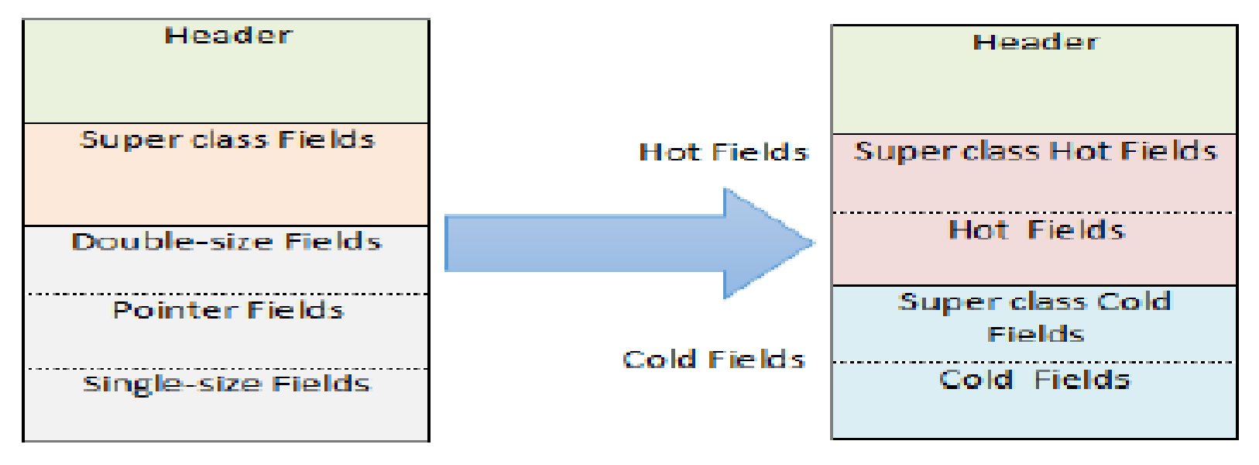
Data layout optimization is a well-known method to improve cache performance by reorganizing data elements. We introduce a novel approach to optimize layout of objects called the Object Splitting Technique. In this approach, Java objects are split at allocation time so that those fields of the split objects that are not accessed as often are separated from the rest of the fields. We implemented the approach in IBM's JVM. The modified JVM was tested with different benchmarks and in most cases the number of cache misses was reduced.

INTRODUCTION

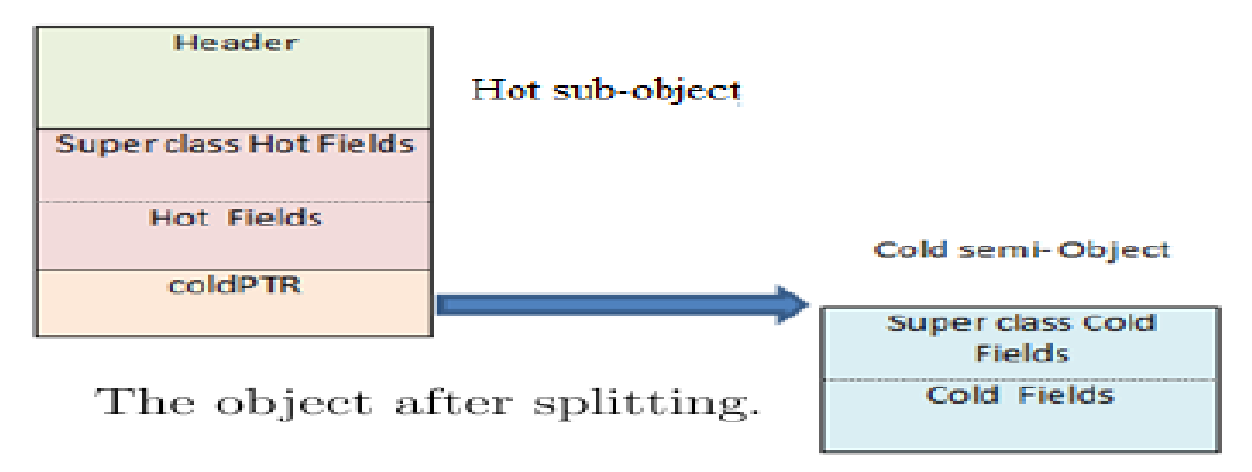
In the Object Splitting Technique (OST), objects are split into two parts at allocation time based on the hotness of their fields.



OST DESIGN



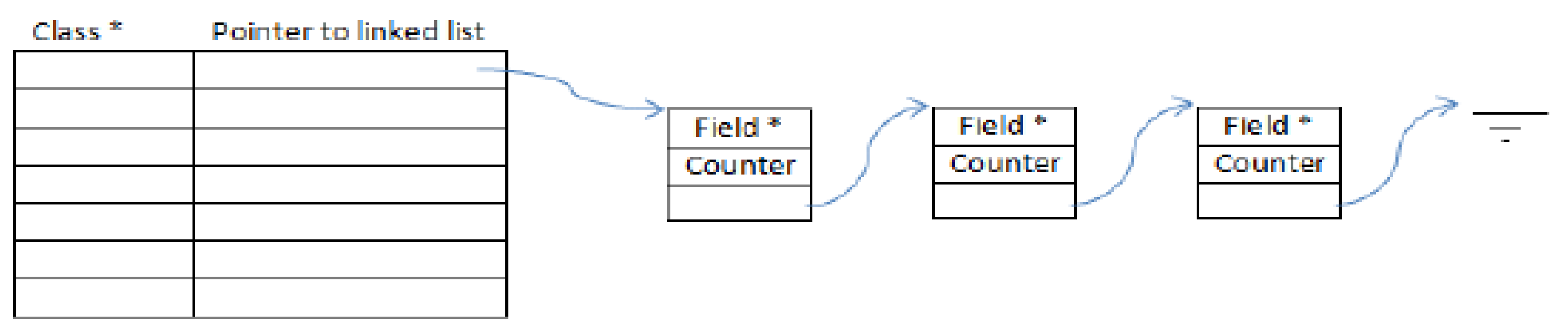
Fields are reordered based on their hotness.



The object after splitting.

PROFILING

In the profiling step, information about hotness of fields and classes is obtained.



Only classes with more than eight bytes of cold parts are selected to be optimized.

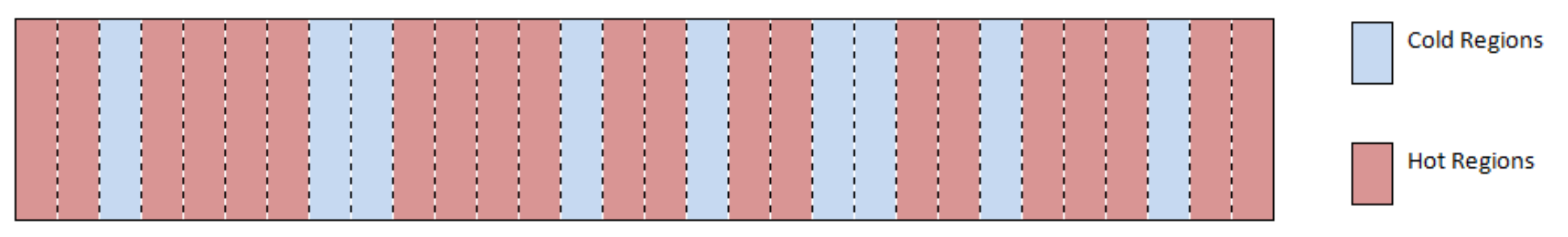
OST IMPLEMENTATION

* The JIT is turned off.

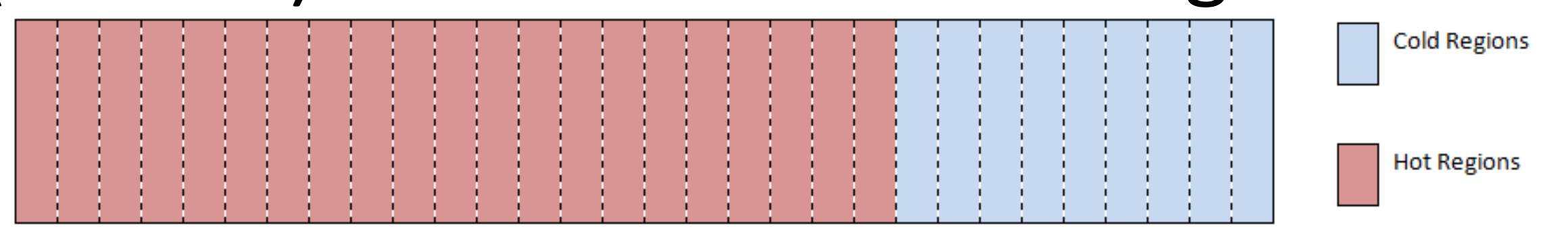
* OST is done with Balanced Garbage Collection.

* Cold parts are allocated in cold regions:

•Region-Based OST (R-OST)

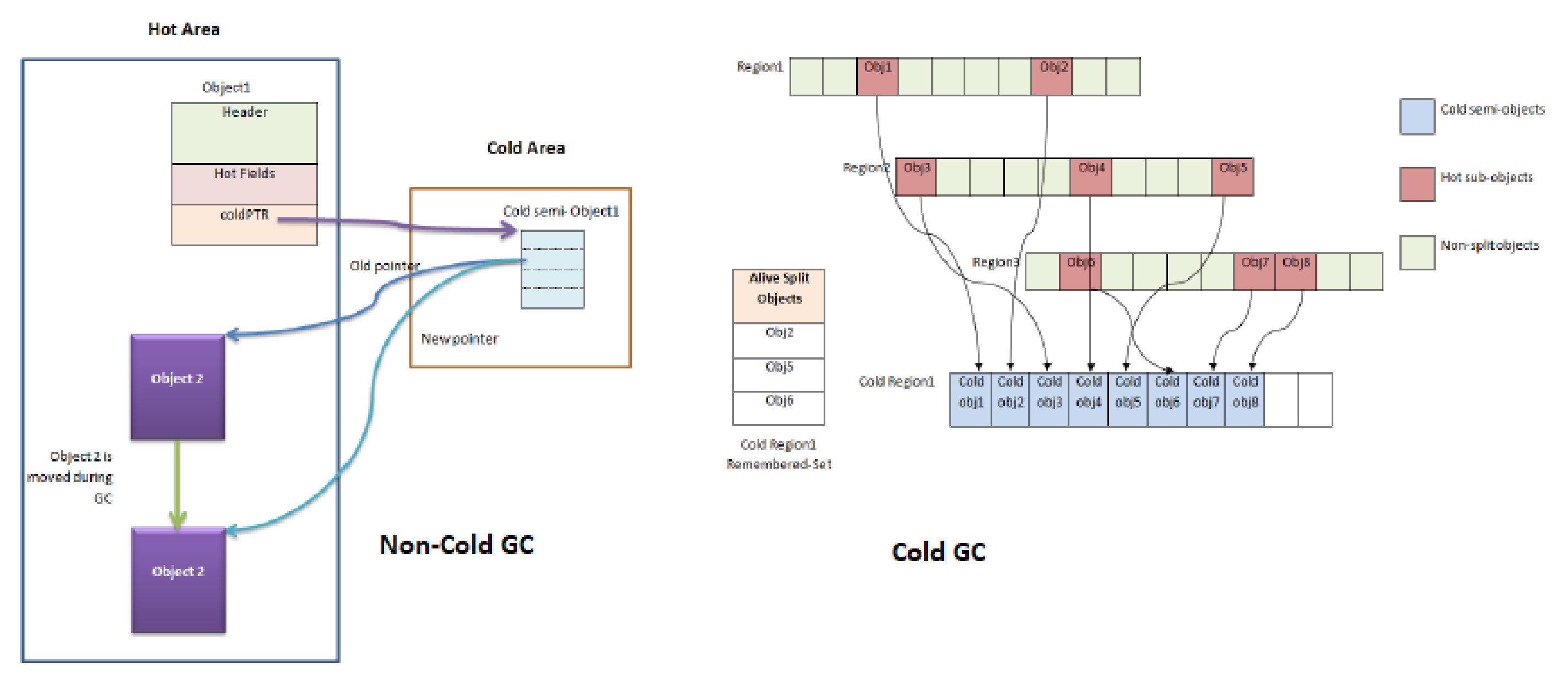


•Allocation Context-Based OST (AC-OST) which takes advantage of the NUMA architecture.

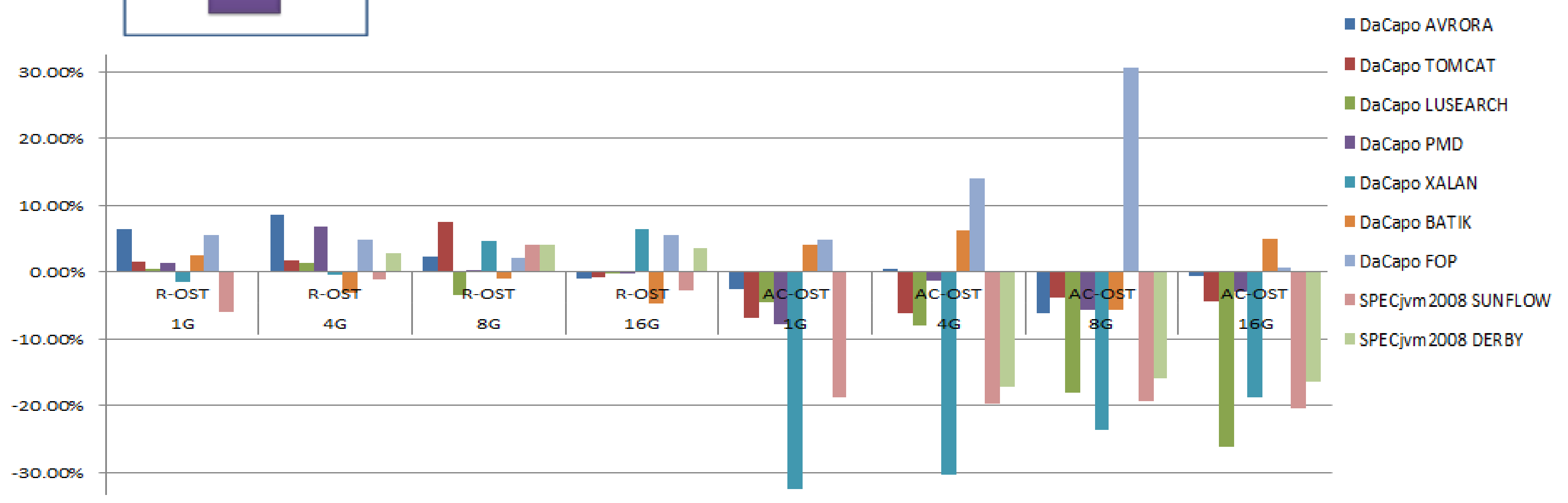


Garbage Collection in the OST

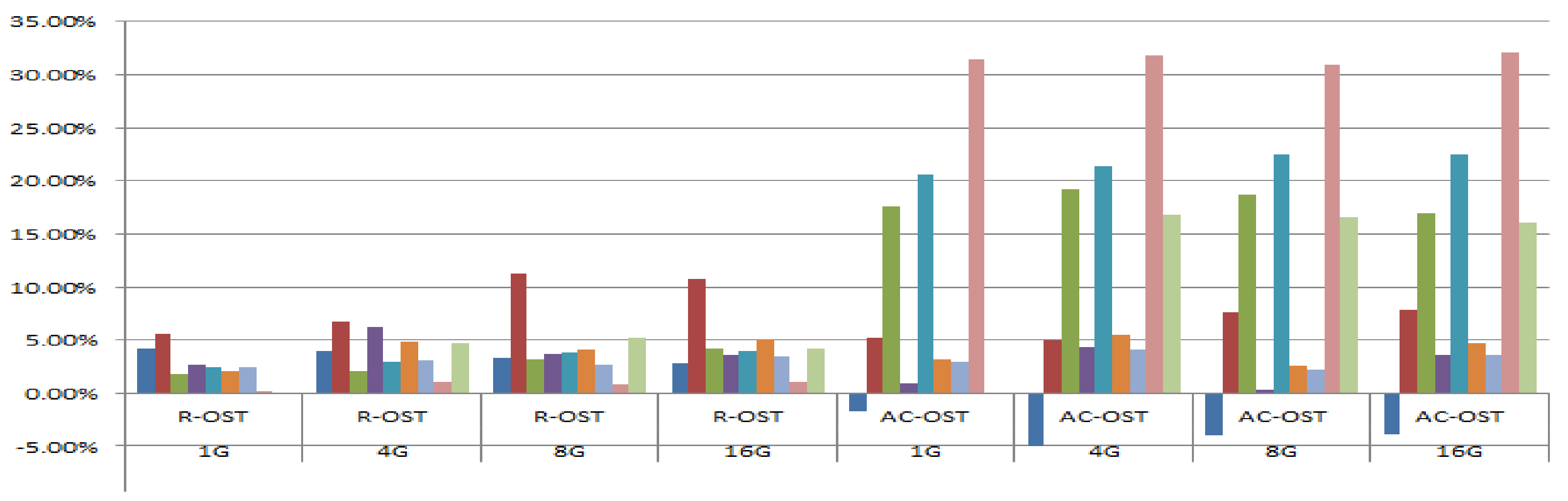
* Regular Partial GC (**PGC**) for non-cold eden regions * Cold Partial GC (**cPGC**) for cold eden regions * Global GC (**GGC**) for non-cold regions *Cold Global GC (**cGGC**) for cold regions



RESULTS



A comparison of the cache performance in different benchmarks (negative numbers show improvement).



A comparison of the execution time in different benchmarks (negative numbers show improvement).

CONCLUSION

- AC-OST can improve cache performance by 30%.
- Turning off the JIT and remote accesses to the cold fields slow down the execution time significantly.