

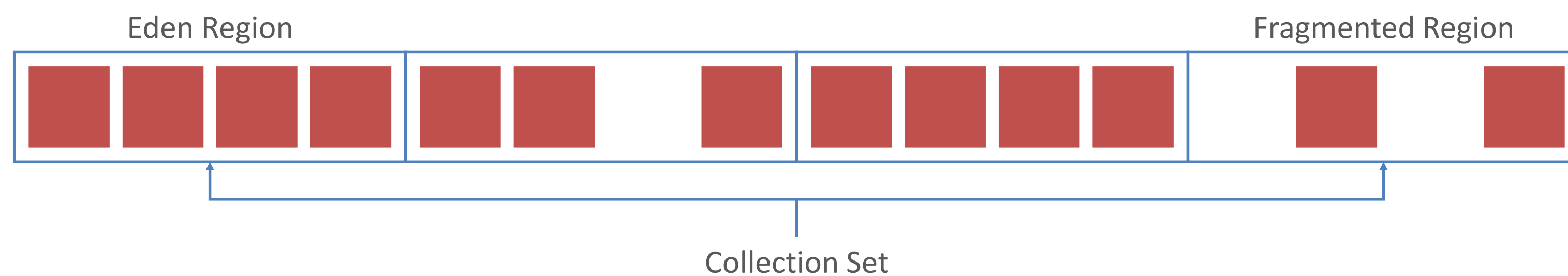
BALANCED COLLECTION SET IN GARCOSIM

Sasha Williams, Kenneth B. Kent, Gerhard Dueck

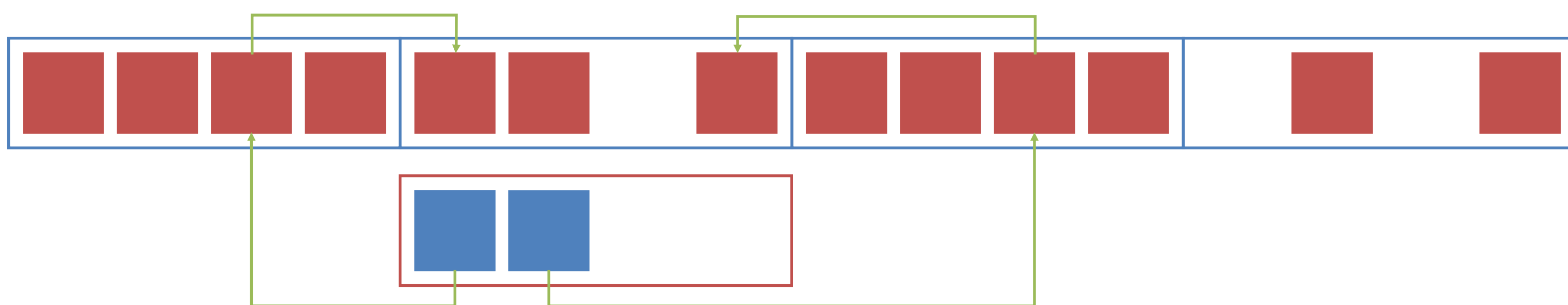
University of New Brunswick, IBM Canada
Faculty of Computer Science
{s.williams, ken, gdueck}@unb.ca

BACKGROUND

Balanced garbage collection attempts to reduce the costs associated with garbage collection by dividing the heap into regions. It leverages the hypothesis that objects die young by collecting regions with newly allocated objects (known as eden regions) as well as highly fragmented regions. The subset of regions to be collected is known as the collection set.



Each region has a remembered set (remset) which remembers references between regions.



PREVIOUS METHOD

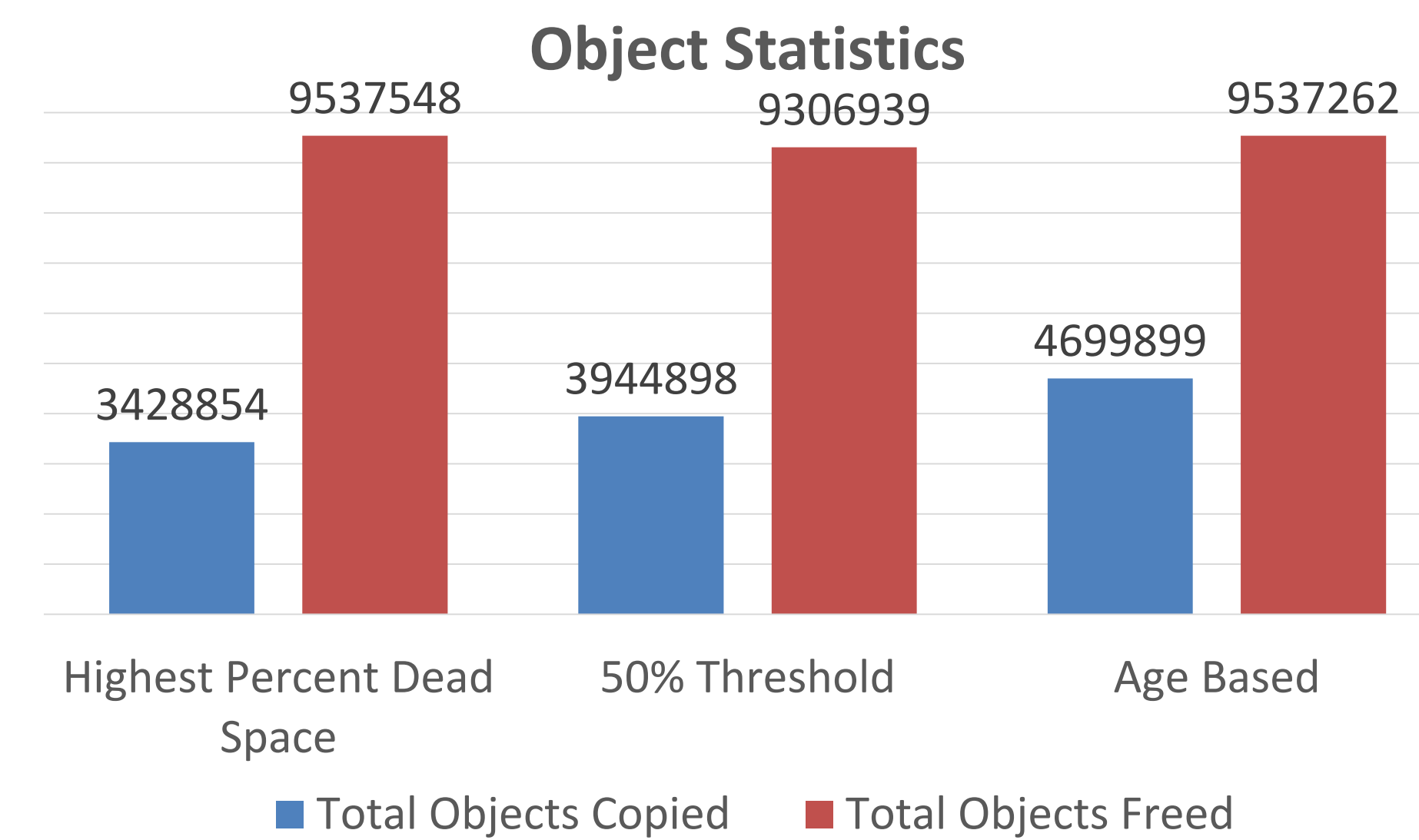
Previously GarCoSim would add all eden regions to the collection set. It would then randomly select the remaining collection set regions based on age; the younger a region the more likely it is selected. A region's age is simply how many times the objects in the region have been copied (up to a maximum age).

NEW METHODS

1. All eden regions and regions containing the highest percentage of dead space
2. All eden regions and regions meeting a minimum percentage of dead space

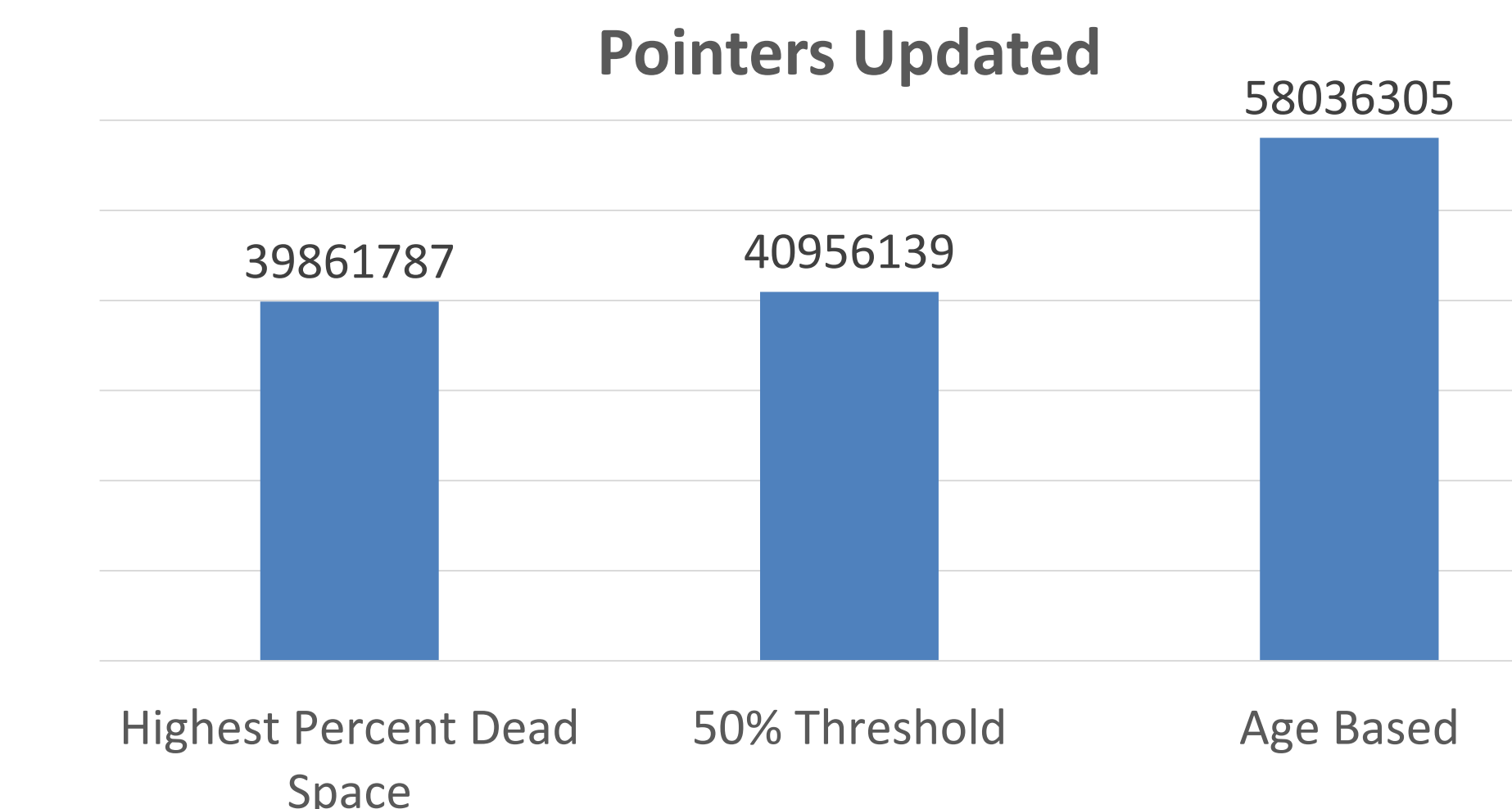
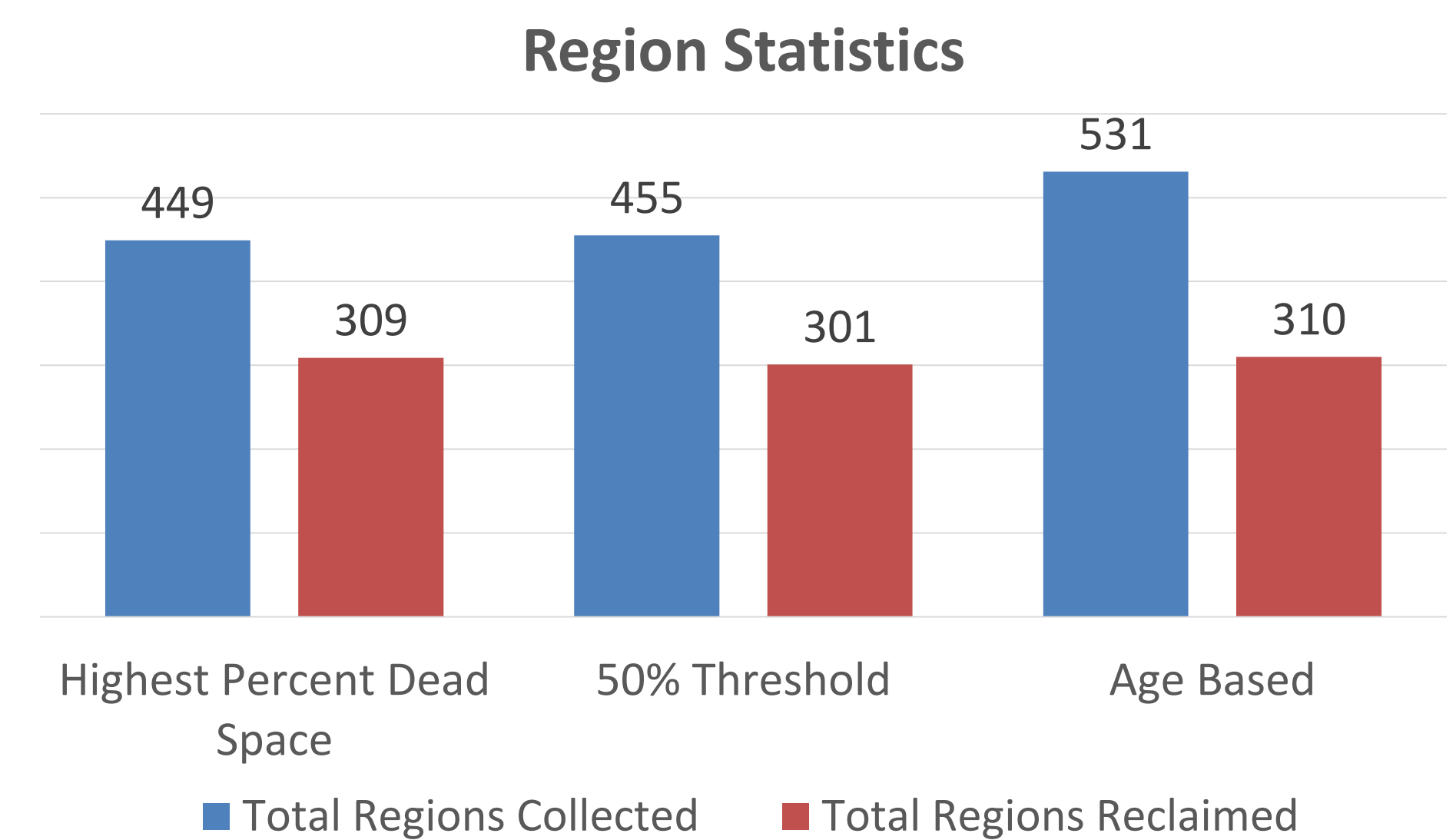
RESULTS

The following results are from the trace file compiler.sunflow and have an initial heap size of 256MB.



Copying more objects takes more time, but freeing more objects means more space for new objects.

Collecting more regions takes more time, but reclaiming more regions means more time until next garbage collection.



Updating more pointers causes the garbage collection to last longer.

New remset entries are made in copy-to regions, while removing and modifying remset entries applies to regions not in the collection set.

The effort to update the remsets was measured by the number of new entries, removed entries, and modified entries. Fewer operations results in a shorter garbage collection time.

