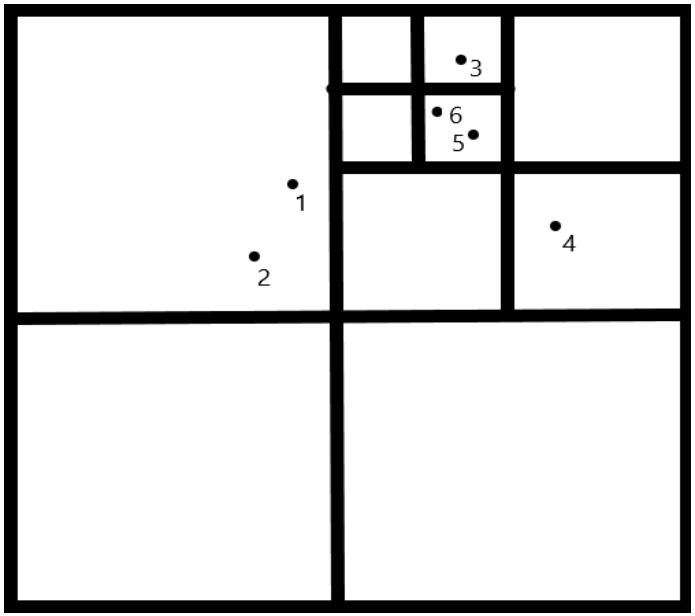
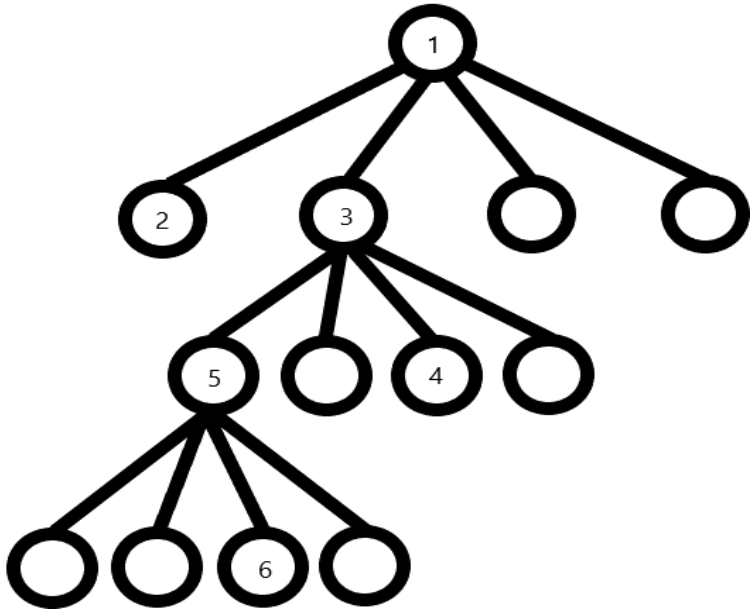
Detector de colisiones

David Calle Daza
201710031010
Medellín, 15 de mayo de 2018

Designed Data Structure

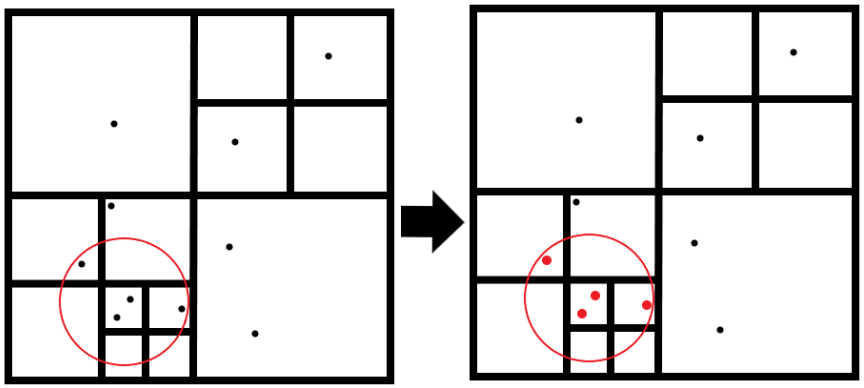


Graphic 1: Geographic representation of the Quadtree



Graphic 2: Abstract representation of the Quadtree nodes

Data Structure Operations



Graphic 3: Function Query by Quadtree

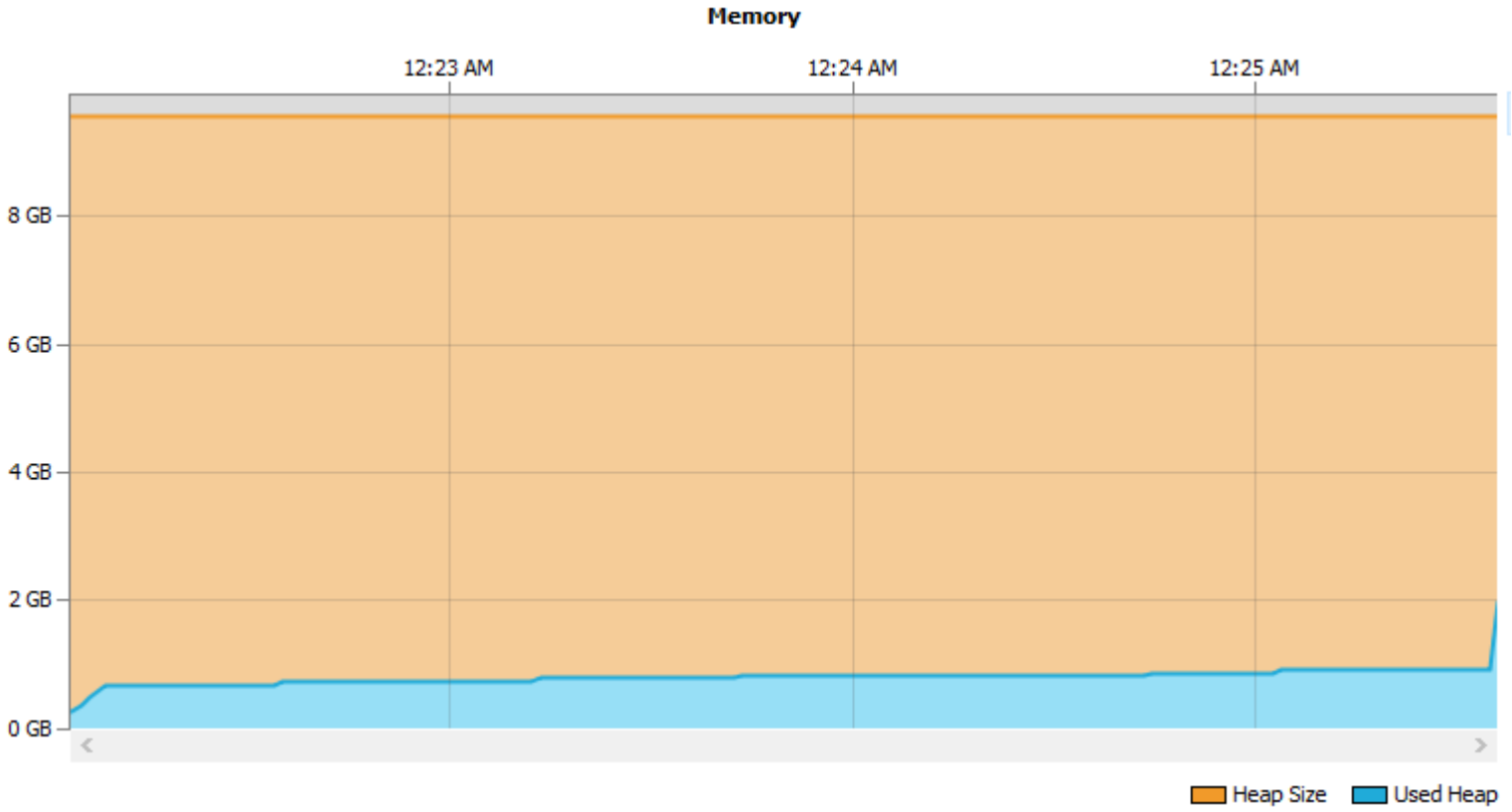
Function	Complexity
Insert	O(log s)
Query	O(log n)
Print	O(n)
Detect collitions	O(nlog n)

Chart 1: Complexity of Quadtree´s functions.

Design Criteria of the Data Structure

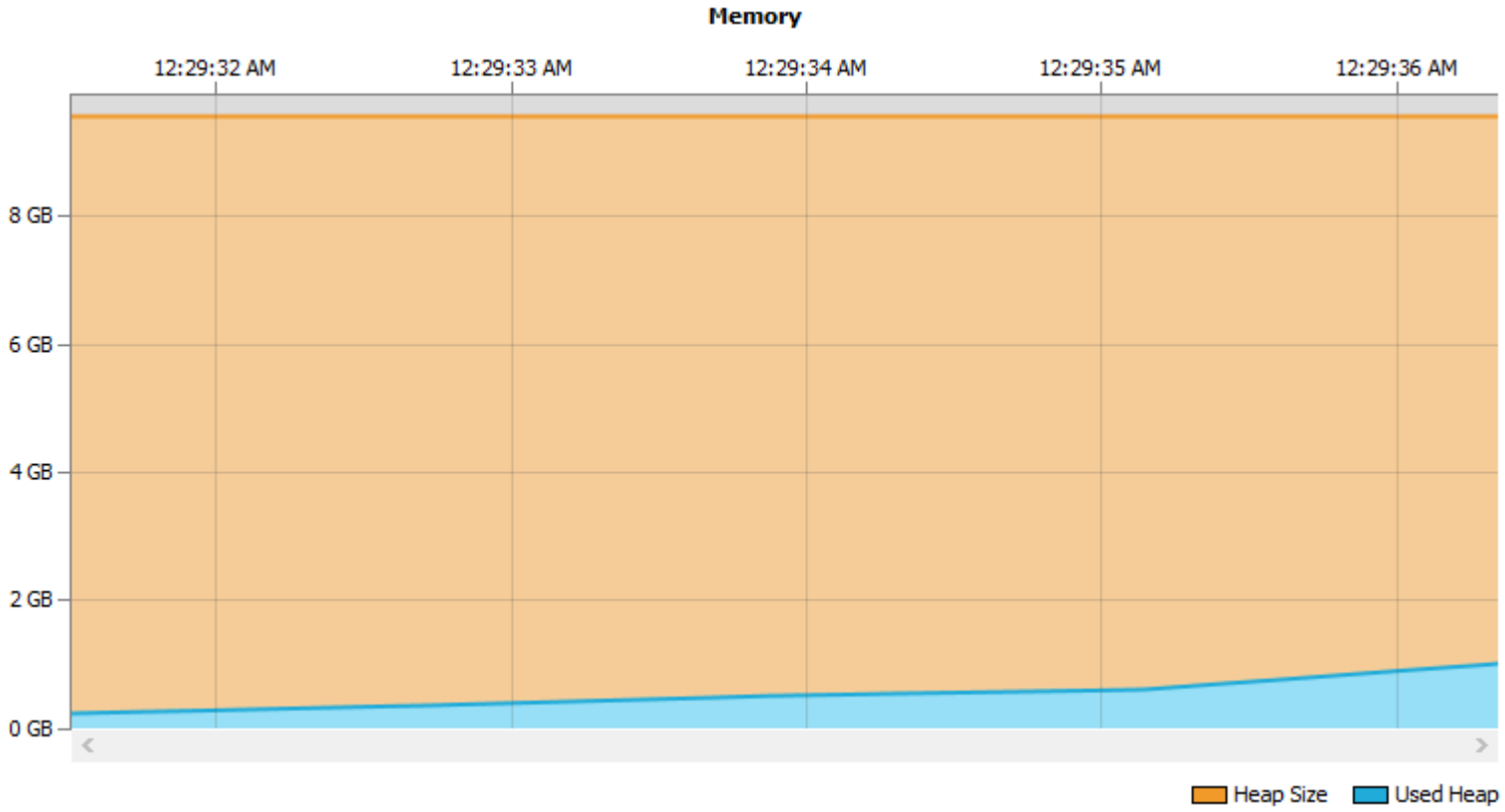
- To solve the problem, we should compare the distance between ALL bees.
- Do this explicitly, would have a complexity of $O(n^2)$.
- The Quadtree divide every bee in individual quadrants, which would simplify the problem.
- ALL Quadtree operations have complexities lower tan O(nlog n).
- Use another kind of tree is lower efficient, therefore, efficient to solve the problem.

Time and Memory Consumption



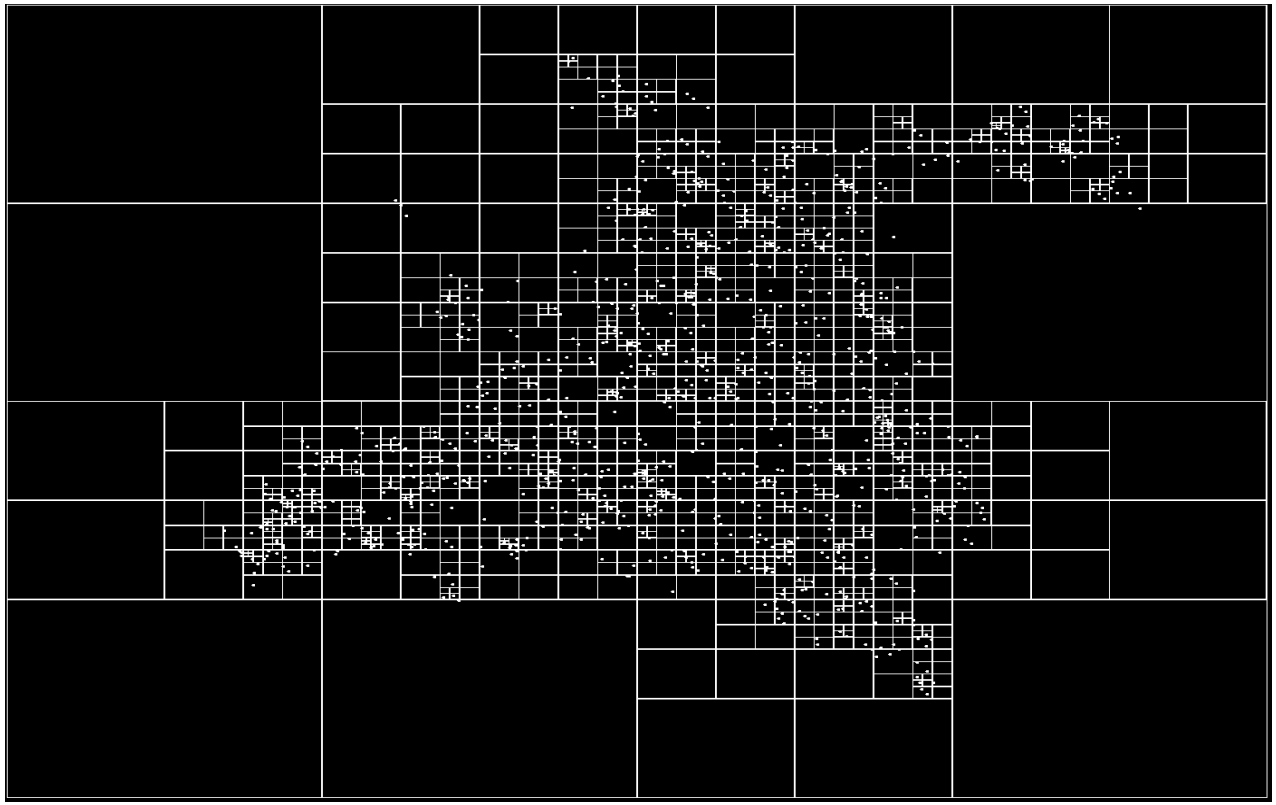
Graphic 4: Total memory and total time using arrays.

Time and Memory Consumption



Graphic 5: Total memory and total time using Quadtree.

Implementation



Graphic 6: Quadtree implementation with 1.000 bees.

Report in arXiv

Ño