Assignment 9 Analysis

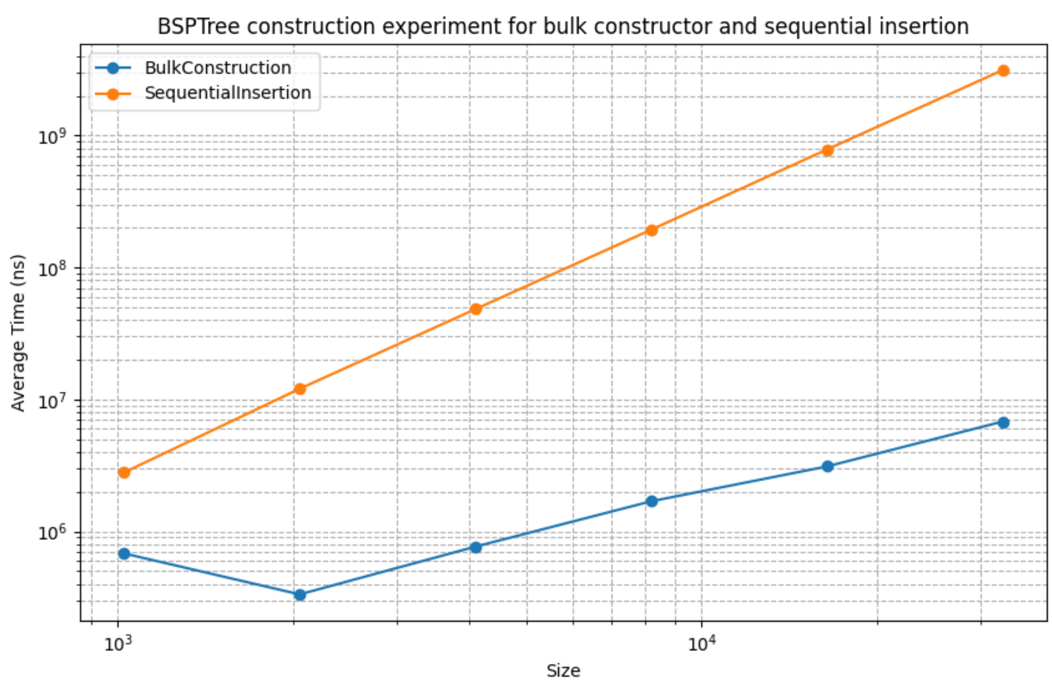
Ray Ding

Construction

What is the "worst case" order to insert them? Plot the runtimes of the 2 methods for building the tree. Do your experiments match the Big O growth rates you expected?

The worst case could be that segments are sorted by their x-coordinate in ascending order. Each new segment could potentially split all the segments that have been inserted before it.

As shown by the figure, the log-log scale plot provided illustrates the relationship between the size of segments and the average time taken to construct a BSP tree using both bulk construction (blue curve) and sequential insertion (orange curve) methods. In a log-log scale plot, a linear relationship indicates a power-law correlation between the two variables. The steepness of the orange curve is consistent with the $O(N^2)$ complexity often associated with the sequential insertion method in worst-case scenarios. The shallower slope of the bulk construction suggests an algorithmic complexity better than $O(N^2)$, potentially close to O(NlogN), which is typically desired for a BSP tree construction. Thus, the results match theoretical expectations: bulk construction should generally perform better than sequential insertion in the worst-case scenario due to its ability to maintain a more balanced tree.

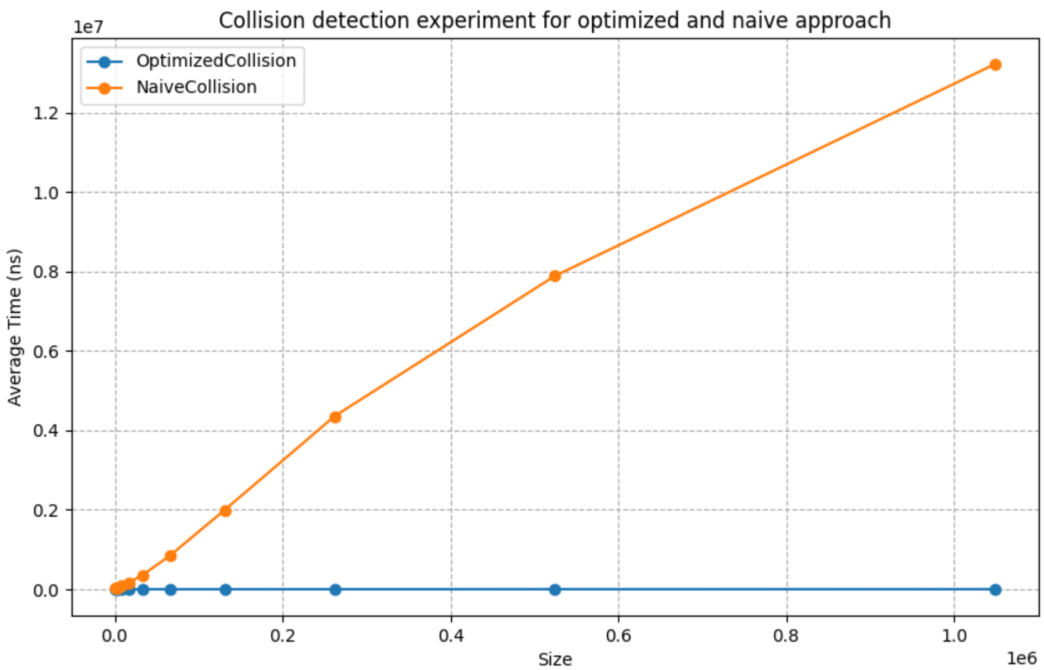


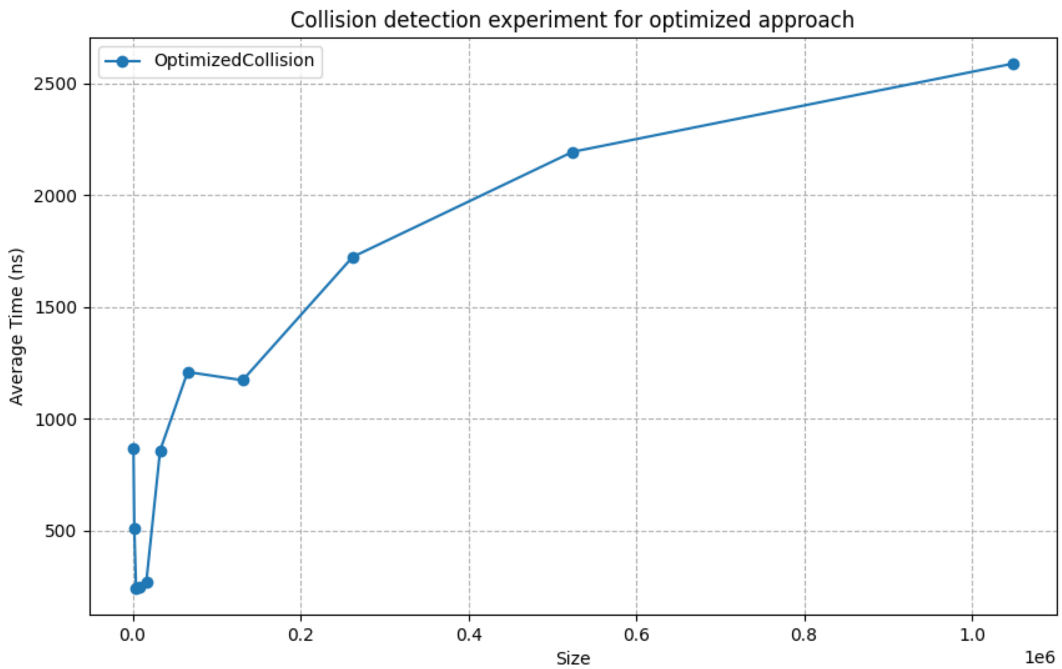
Collision Detection

Does our optimized collision detection routine run in the big O you expected? Be sure to describe the details of your experiment.

Yes, the routine behaves as expected.

The expected Big O complexity for the optimized collision detection routine in a well-balanced BSP tree is O(logN), where n is the number of segments in the tree. This is because the collision detection should only need to check a subset of segments along a path from the root to a leaf. The naive method has an expected complexity of O(N) since it checks every segment.




Collision detection experiment for optimized approach
OptimizedCollision
Average Time (ns)
2500
2000
1500
1000
500
0.0
0.2
0.4
0.6
0.8
1.0
1e6
Size