Khoi Trinh

DSA 5005

Project 6 Report

Dec 3 2022

1. **Introduction**

The purpose of this report is to showcase a comparison between 2 sorting algorithms: bubble sort and shell sort. While both can produce a completely sorted array of numbers, their speed differs greatly. Moreover, if time is a constraint for a sorting algorithm; as in; if the algorithm is only allowed to do D number of comparisons, which one would outperform the other?

The 2 metrics that are used to compare these algorithms are: the number of inversions; and the Chebyshev’s distance. Below is a brief explanation of these metrics: (these were taken from the project specifications)

Inversion: An inversion in a given random set of numbers σ is a pair (σ(i),σ(j)) such that i < j and σ(i) > σ(j).   
Where i and j are index values and σ(i) and σ(j) are the numbers at those positions.

Chebyshev’s distance: Given random set of numbers σ1, and the same set of numbers completely sorted σ2, the Chebyshev distance (dl) between σ1 and σ2 is

dl∞ (σ1,σ2) = |σ1 −σ2|∞ = max |σ1(i)−σ2(i)|.

1. **Experiment Setup and Results**

For this experiment, the algorithms are ran on a number of different arrays of randomly generated numbers. These arrays are 1000, 5000, 1000, 20000, and 30000 randomly generated elements. The number of comparisons D for each algorithm are as follows: n, 2n, 4n, 10n, 50n, 100n, 200n, 1000n, 2000n, 4000n, n2 with n being the number of elements in the array. For example; for n = 5000; D will be 5000, 10000, 20000, 50000, 250000, 500000, 1000000, 5000000, 10000000, 20000000, and 2500000.

For the following tables; the rows are:

Bub In: number of inversions for bubble sort

Bub dist: Chebyshev’s Distance for bubble sort

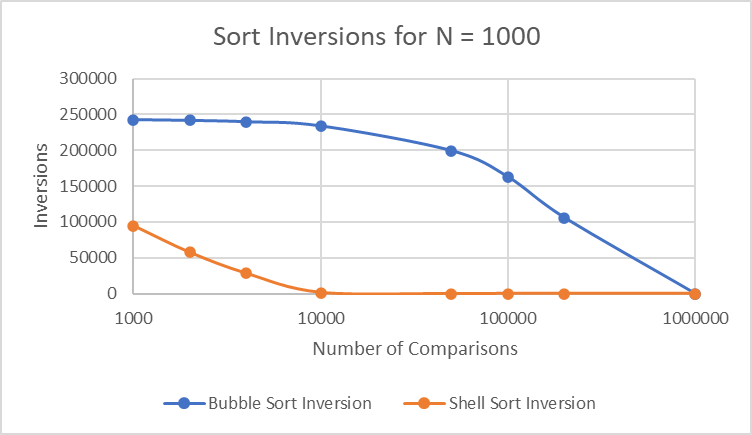
Shell in: number of inversions for shell sort

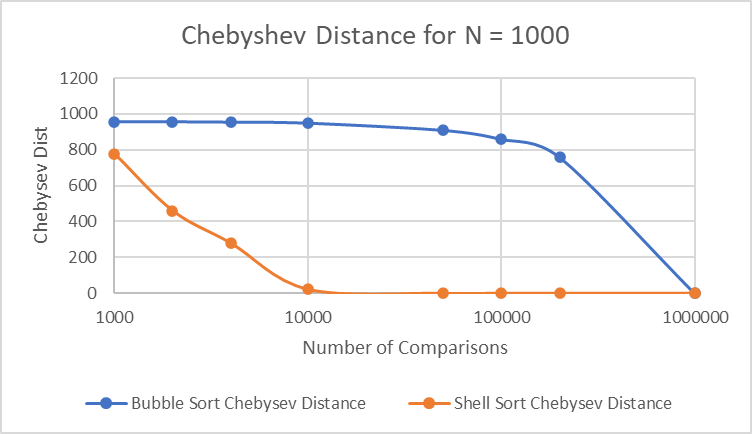
Shell dist: Chebyshev’s Distance for shell sort

First, for 1000 elements; the results are as follows; note that due to 1000n = n2 for this case; the table are smaller.



Graphically; note that the x-axis is log scale base 10

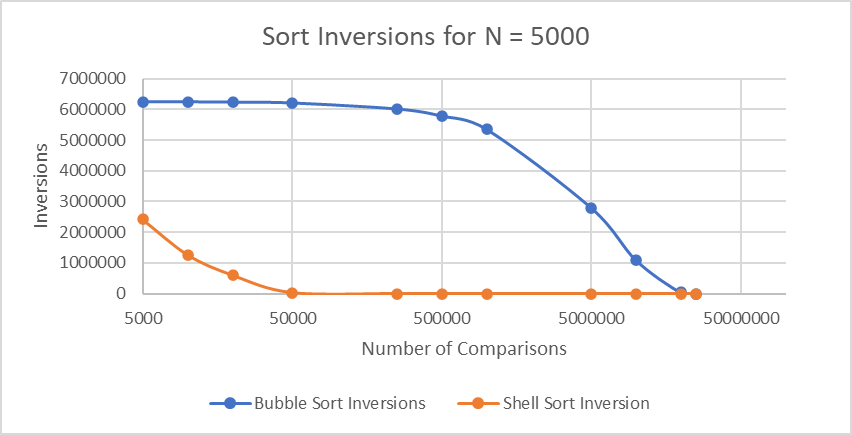


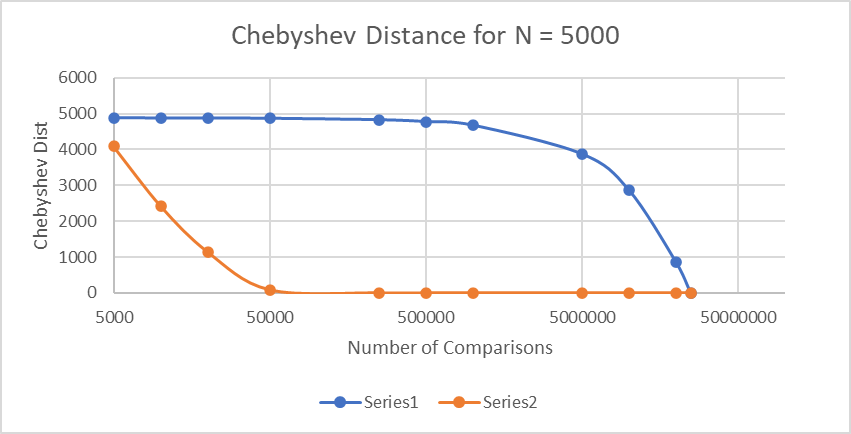


Next, for n = 5000.



Graphically:





Next, for n = 10000.



Graphically:

Next, n = 20000.

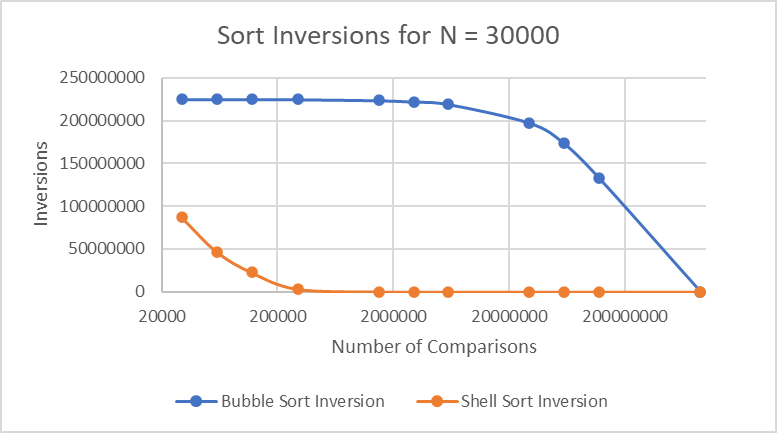


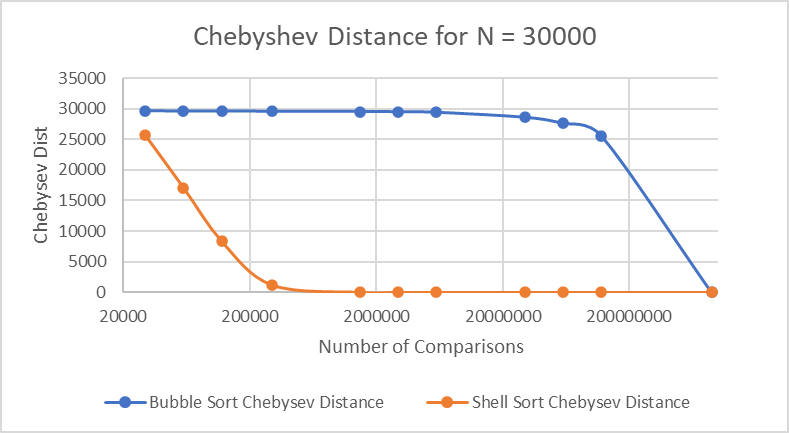
Graphically:

Finally, n = 30000



Graphically:





1. **Analysis and Conclusion**

From the graphs and tabulated results; a clear pattern can be seen. For shell sort, the number of inversions and Chebyshev Distance reaches 0 when D = 50n; meaning that after 50n comparisons, shell sort algorithm will completely sort an array. On the contrary, bubble sort inversions and Chebyshev Distance don’t reach 0 until D = n2; which is much longer than shell sort. In conclusion, given the same number of comparison, shell sort would be a better choice for a sorting algorithm.