CS 310
Assignment 321

Arjan Regmi
ar4752

For the class in this analysis, we implement the brute force and divide and conquer methods to calculate the shortest distance between any two points from the input file, which is a set of unique n points.

For both methods, the value for n is taken to be the input size, that is, a set of n unique points. We use the same input for both methods as we are trying to analyse the two algorithms.

For the empirical analysis we measure the number of basic operations performed to calculate the shortest distance for a given input size. To analyze the brute force mechanism to compute the shortest distance between any two points, we chose the computation of square root of the difference of the x-coordinates and y-coordinates of any two pairs, on line 55 as our basic operation.

To analyze the divide and conquer mechanism to compute the shortest distance between any two points, we chose the division of n coordinates into two sets in lines 89 and 96 and also the computaion of the shortest distance between the two pairs on line 134 as our basic operations.

An empirical analysis of the brute force mechanism running the algorithm for multiple values of n produces the results shown below. Standard functions $f(n) = n^2$ and $f(n) = 0.5 * n^2$ are also shown.

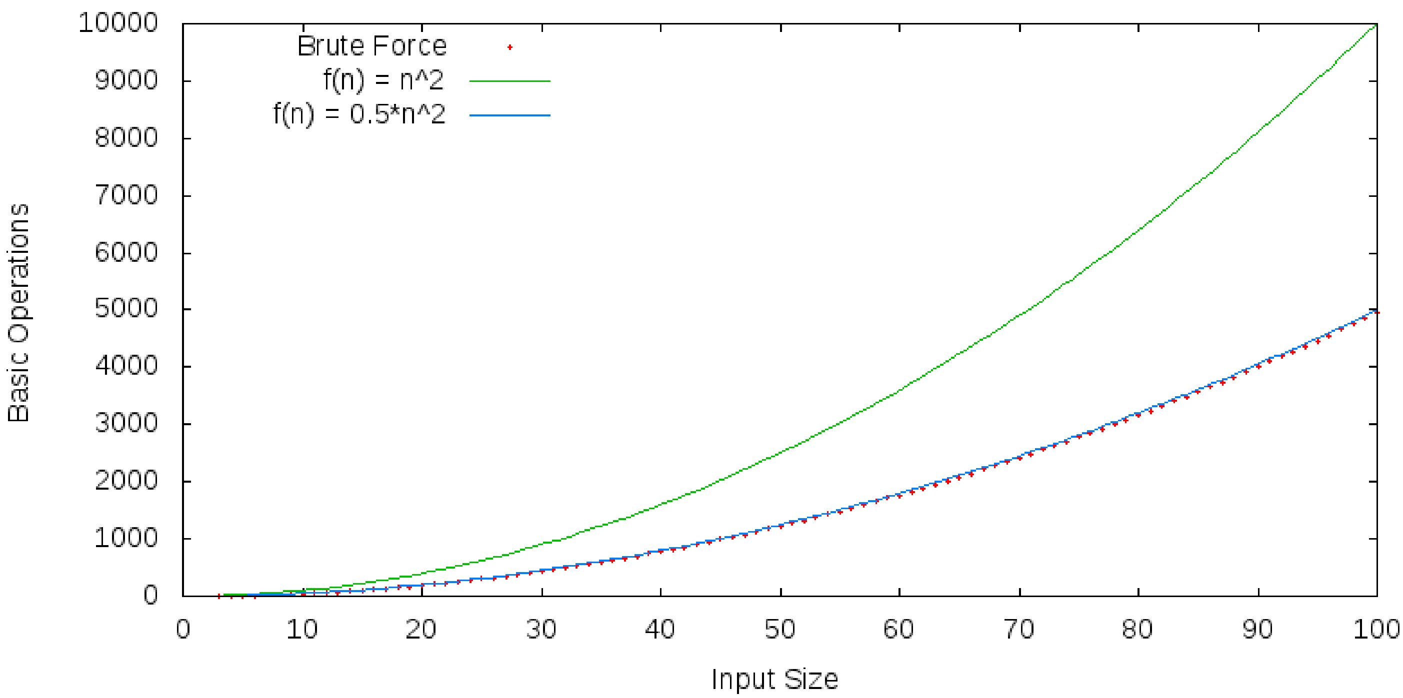

Figure: Basic Operation vs Input Size (Brute Force Algorithm)

An examination of the code itself explains the empirical results when we observe that the brute force algorithm shows a quadratic nature when plotted for multiple values of n. Here, we obtain an asymtotically tight bound with the standard function n^2. Also, this algorithm cannot end early as it compares the distance of each pair of points with every other pair and computes the shortest distance.

Therefore, we conclude that the brute force algorithm is described by

$$T(n) \in \Theta(n^2)$$

Secondly, an empirical analysis of thedivide and conquer mechanism running the algorithm for multiple values of n produces the results shown below. Standard functions $f(n) = nlog(n)$ and $f(n) = n^2$ are also shown.

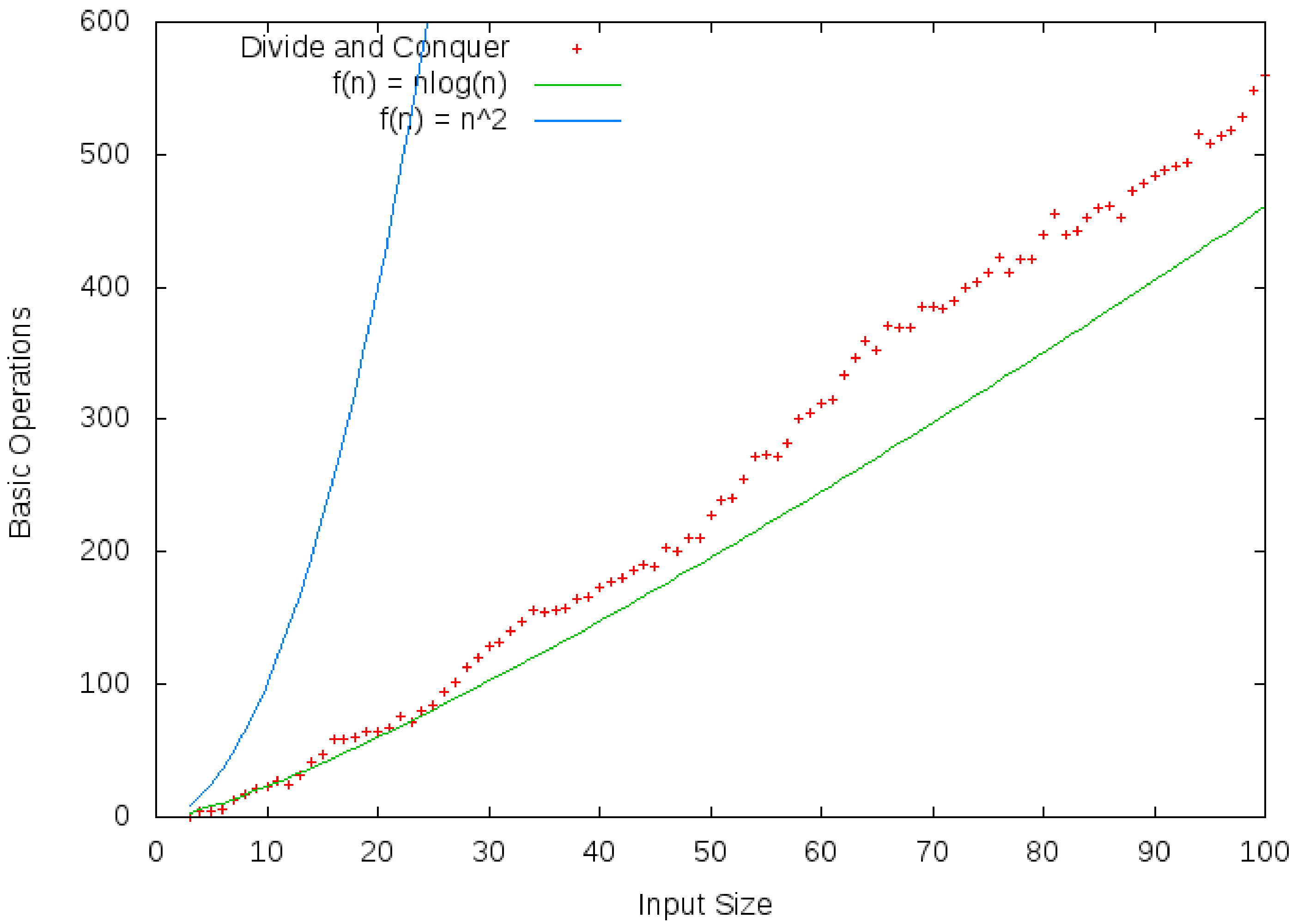

Figure: Basic Operation vs Input Size (Divide and Conquer Algorithm)

For the analysis of divide and conquer, the above figure shoes that the standard function $nlog(n)$ upperbounds the number of basic operation as the input size increases. We know from the code, the divide and conquer algorithm divides all the points into two sets ($log(n)$) and recursively calls for two sets. After dividing, it finds the strip in $O(n)$ time. Also, it takes $O(n)$ time to divide the Py array around the mid vertical line.

With the help of the graph above we can concludethat the divide the conquer algorithm has an analysis of:

$$T(n) \in O(nlog(n))$$