

Closest Pair Report

Max Andreas de Visser, Agne Sutovaite, Casper Anton Poulsen,
Sebastian Rohr, Thomas Fosdam Claudinger

September 21, 2022

Results

Our implementation is written in Python and produces the expected results on all input-output file pairs, except d2103-tsp.txt, where our code reports distance 20.1121 as the shortest distance and the correct answer were 20.01724. This may be because of rounding errors, or differences between float and double on various machines. When looking at the scatter plot of the points in Figure 1, we see that it the points looks very odd. We therefore suspect that it is not a programming mistake, but another kind of error. Giving that this is close to being the worst case scenario for a divide and conquer algorithm, getting that close to the correct answer, we'd say is alright.

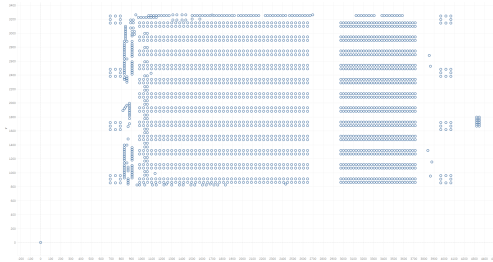


Figure 1: Scatter plot of points in d2103-tsp.txt.

Implementation details

We resort by y -coordinates in each recursive step.

For the comparison of points close to s in S_y we inspect 11 points, as explained by Thore Husfeldt in Algorithm Design lecture 3 (slide 46). Here is the corresponding function from our code:

```
def closestInSlice(points: list, n: int, minimum):
    minimum_ = minimum

    for i in range(n):
        for j in range(i+1, min(11,n)):
            if dist(points[i], points[j]) < minimum_:
                minimum_ = dist(points[i], points[j])

    return minimum_
```

Our running time is $O(n \log^2 n)$ for n points, since we resort it at every recursion step.