CS 345A Assignment 1

Yasharth Bajpai (170822), Nidhi Hegde (180472)

September 16, 2020

Difficult

Overview:

As suggested by the question, our algorithm reduces the running time for finding all non dominated points in a given set of points from $O(n \log n)$ to $O(n \log h)$ where h is the number of Non-Dominated points in the set. To achieve this we took a hint from the first approach discussed in the lecture where we had swept through the points with maximum X-coordinate meanwhile deleting the points that were dominated by this point.

Similarly here, we pick out the point of maximum Y-coordinate in the right half of every subproblem and go on to remove all the points dominated by it before sending it onto further recursion. As we keep on deleting the points that would end up being dominated, we can say with certainty that, every point which emerges with a maximum y coordinate in the right half of a subproblem(or the problem) is a Non-Dominated point not only for the sub problem, but also the Problem that we began with.

At every subproblem we reach, we find at least one Non-Dominated point(of the largest problem), so to cover the entire set of h non Dominated points, we only need to access h such problems. Thus bringing down the depth of the recursion tree from $\log n$ earlier(when we recurred down to all n points at the leaf) to $\log h$. Therefore, helping us complete the entire process with a possibly greater efficiency.

Algorithm 1 To Find all Non-Dominated points in $O(n \log h)$

function *nonDominated*(S)
    $P_{max\,X} \leftarrow Xmax\,(S)$
    $P_{max\,Y} \leftarrow Ymax(S)$
    if ($P_{max\,X} == P_{max\,Y}$) **then**
        $S = S$ - dominatedBy($P_{max\,Y}$)
        return $\{P_{max\,Y}\}$
    Compute X-median of S
    $(H_L, H_R) \leftarrow$ Split by X-median(S)
    $P \leftarrow Ymax(H_R)$
    $S_L = H_L$ - dominatedBy(P)
    $S_R = H_R$ - dominatedBy(P) - P
    return *nonDominated*(S_L) $\bigcup \{P\} \bigcup$ *nonDominated*(S_R)

Correctness:

Divide Step: We divide the sub-problem(or problem) into 2 sub-problems based on the *x median* of the set, find the point with maximum *y coordinate* from the right sub-problem, $P_{R\ max\ Y}$ and remove all the points dominated by it in the left sub-problem as well as right sub-problem.

Conquer/Combine Step: During the conquer step, we collect the $P_{R\ max\ Y}$ obtained in the current context, the non Dominated Sets for the left & right sub-problems and then take a union of all three to get the non-Dominated Set for the problem.

Tree Structure:
We model the solution using a tree wherein a given sub-problem is represented by a node within the tree. Also, there is exactly 1 non-dominated point associated with each node of the tree. We define this point to be the point with maximum y co-ordinate in the right half of the the node. We process the given node according to this point, i.e. remove all the points dominated by it within the left and right half of the node and subsequently proceed to the next level of the tree, marking the left and right halves as the children of this node.

Some observations:

1. If a point is non-dominated in a set P, then it will be non-dominated in any subset S of P.
2. If a point is dominated in any subset S of P, then it will be dominated in the entire set P.
3. If p is a point that dominates a given set of points P, then any point which dominates p will also dominate all the points within the set P.

Proof by Induction:

Induction Hypothesis: A point non-dominated within any node of a level will be non-dominated within all its parent nodes, assuming we follow the Divide strategy mentioned above at every level.

For the purpose of the proof, let us understand the notation used below.

1. $S^{\,i}$ denotes any node at the i^{th} level of the recursion tree.
2. For any node $S^{\,i}$, the sets H_L^{i+1} & H_R^{i+1} denote the left and right sub-halves of the node $S^{\,i}$, split using X median.
3. For any node S^i, the point P^i denotes the point with the maximum Y coordinate in the right sub-half of S^i i.e. $P \leftarrow Ymax(H_R^{i+1})$.
4. For any node $S^{\,i}$, the sets $S_L^{\,i+1}$ & $S_R^{\,i+1}$ denote the left and right child of the node S^i, after processing of H_L^{i+1} & H_R^{i+1} for points dominated by P.
   (Any symbol with a superscript is just used as a way to represent its level in the recursion tree)

Level 0:

We start with the entire set of points S^0 and divide it into 2 halves, H_L^1 and H_R^1, based on the median. Let the point with maximum y-coordinate in the right half be P^0 .

Let's say there is a point p' in the set S^0, which dominates P^0. If at all such a point exists, both y-co-ordinate and x-co-ordinate of p' should be greater than that of P^0.

That means no point to the left of P^0 can be p', since its x co-ordinate will be always less than x co-ordinate of P^0. Similarly, no point to the right of P^0 can be p' since its y co-ordinate will be always less than y co-ordinate of P^0. Thus, P^0 is non-dominated in the entire set (i.e. S^0).Hence, the above invariant holds for level 0 of the tree.

Now, we remove all the points dominated by P^0 from H_L^1 and H_R^1 to get S_L^1 and S_R^1 respectively and recursively apply the algorithm on them. The nodes S_L^1 and S_R^1 form level 1 of the tree with 2 nodes.

Level i+1 :

Let us assume that the given invariant holds for all the nodes on a Level i of the recursion tree (Induction Hypothesis is true for level i).

Without loss of generality, let us consider any node S^i at the ith level of the tree. Let the non-dominated point corresponding to this node be P^i.

To complete the proof, we need to show that the invariant also holds for its children, S_L^{i+1} and S_R^{i+1} obtained by the Divide strategy.

First, we process the node S_L^{i+1}. Let the non-dominated point corresponding to this node be P_L^{i+1}. Clearly, P_L^{i+1} is a non-dominated point in the set S_L^{i+1}. Next, since we already removed all the points dominated by its parent P^i in the previous step, P_L^{i+1} cannot be dominated either by its parent or by any of the points removed (by observation 3).

Since P_L^{i+1} is non-dominated by its parent and lies to the left of it, $y_{P_L^{i+1}}$ should be greater than y_{P^i}. And since by definition, P^i is the point with maximum y co-ordinate in S_R^{i+1}, $y_{P_L^{i+1}}$ has to be greater than the y co-ordinate of every element in S_R^{i+1}. Hence, P_L^{i+1} will be a non-dominated point in its parent node S^i.

Next, we process the set S_R^{i+1}. Proceeding similarly, the non-dominated point corresponding to this node will be non-dominated among all the points in S_L^{i+1} since the x co-ordinate of all the points in S_L^{i+1} will be less than its. Hence, P_R^{i+1} will also be a non-dominated point in its parent node S^i.

Thus, we proved that the non-dominated points corresponding to each of the children nodes are also non-dominated within the parent set S^i. And since any point non-dominated in S^i is non-dominated within the entire set, we prove that the invariant holds at level i+1.

Hence, By Induction we prove that a point non-dominated in any Subset(node) S is non-dominated within the entire set.

Base case:

The base case is when a single point dominates all the other points within the sub-problem. This will be true when the the point with the maximum x co-ordinate is the same as that with maximum y co-ordinate. Hence, this point will be added to the list and all the remaining points will be removed.

Time Complexity:

Processing at each level (finding the median, points with Maximum X & Y coordinates and removing the dominated points) all are achieved in $O(n)$. The construction of the solution ensures there is essentially one Non Dominated point associated with each node of the Recursion Tree. The depth of the recursion tree will thus be $O(\log h)$.

Therefore a computation of $O(n)$ takes place at $O(\log h)$ levels, making the resulting algorithm have a complexity of $O(n \log h)$.
(We were unable to obtain an explicit mathematical recurrence since every subsequent recursion depends on the distribution of input set)