NSGA-II Algorithm

# A. Non-Dominated Sorting

For each solution (population member) we calculate 2 entities:

1. Domination count np, the number of solutions which dominate this solution (p)
2. Set of solutions (Sp )that this solution (p) dominates

This calculation has O(MN2) complexity *[M is number of objective/fitness functions]*

(Non-dominated front is a set of solutions that do not dominate each other.)

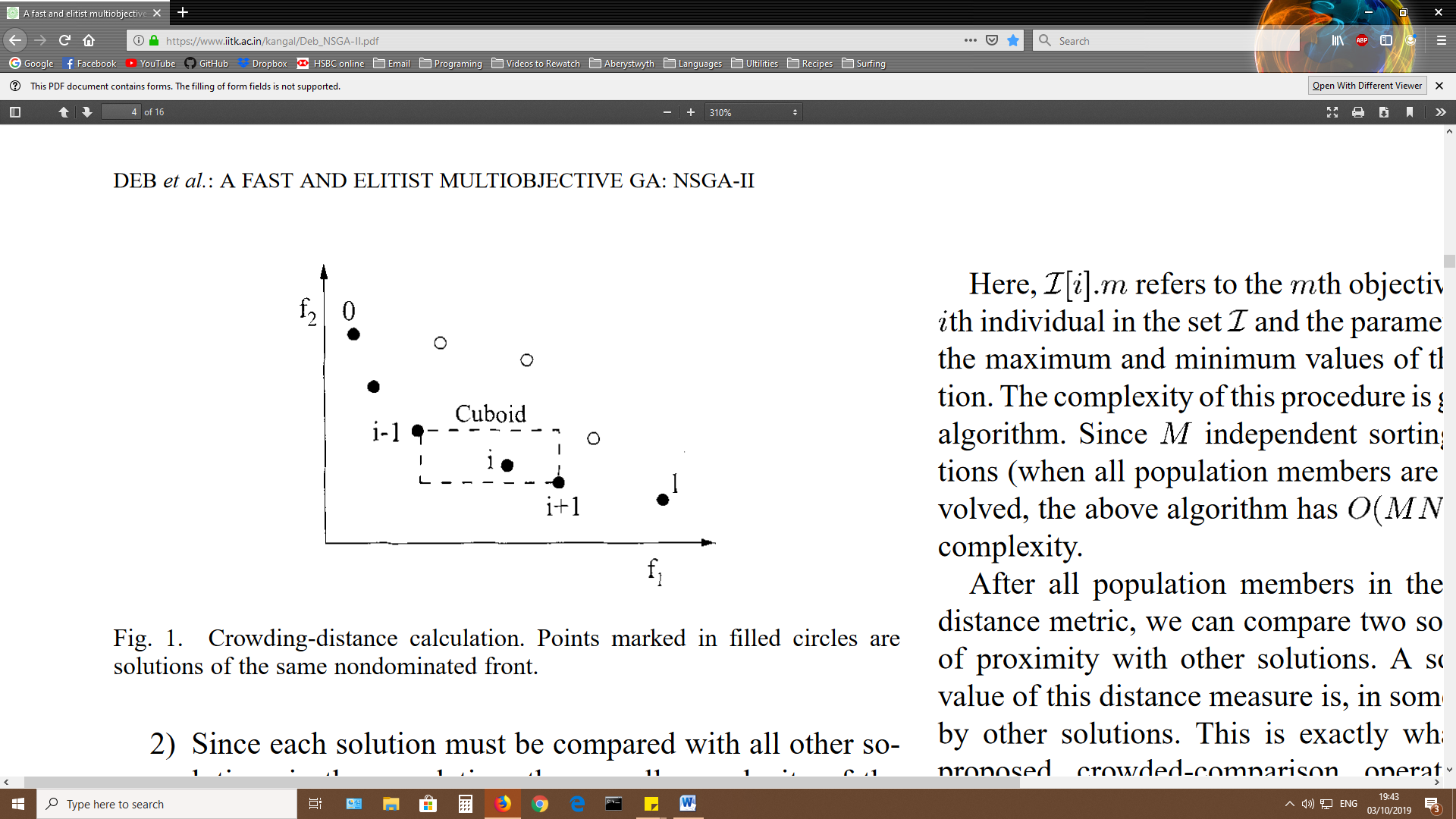
The first non-dominated front will contain the solutions with np =0

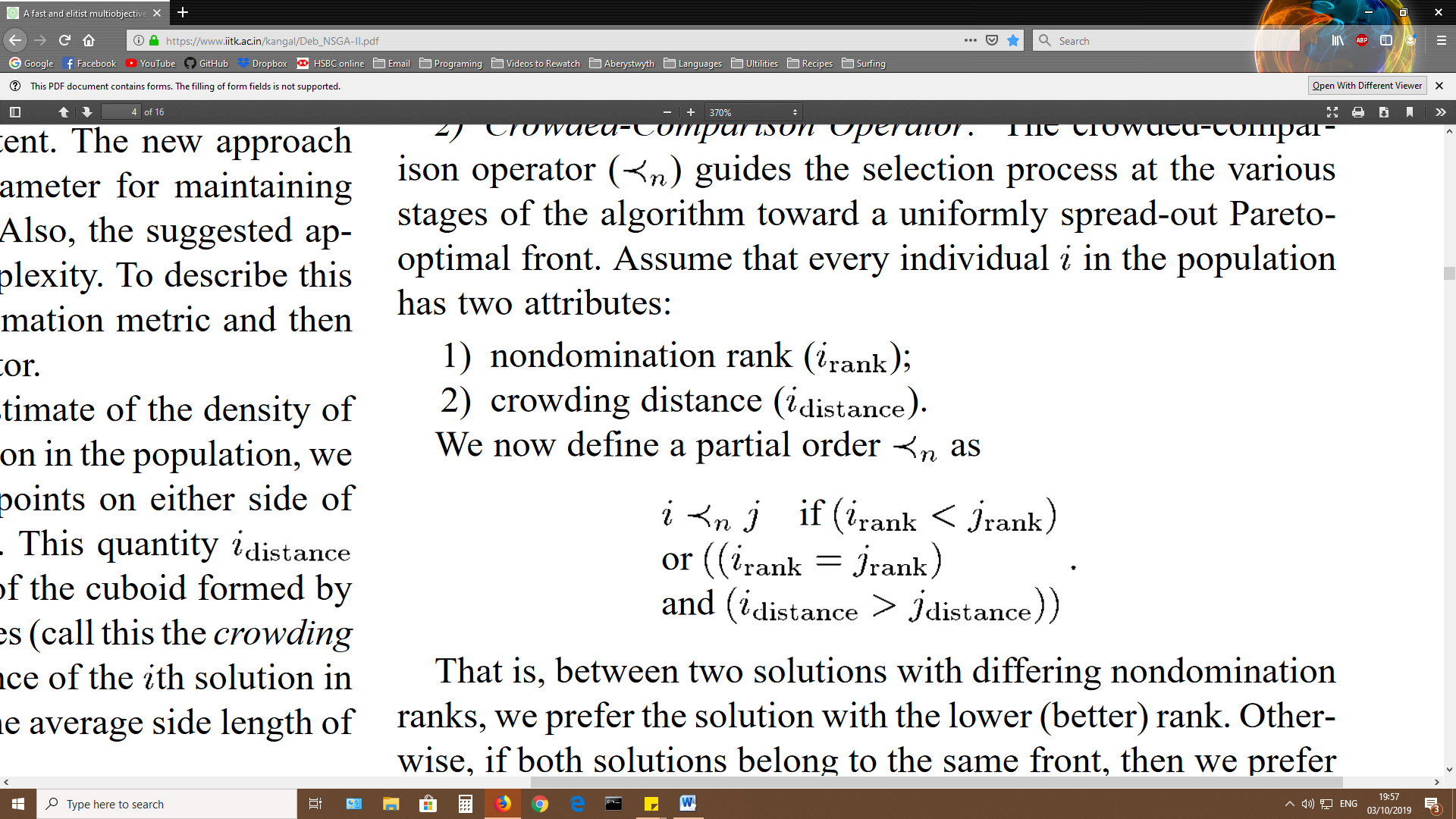
We go to each member (p) of the first non-dominated front and we iterate through each of their dominated solutions (Sp) reducing each of the dominated solutions (q) domination count by one. If the domination count becomes 0 then we put it in a separate list Q. This list forms the second non-dominated front.

The above procedure is repeated until the third front is identified and then until all fronts are identified.

# B. Diversity Preservation

Preserve diversity by comparing crowding distances. Crowding distance is the estimated perimeter of a box/cuboid drawn around a point with the two nearest neighbours (left and right) being used as the vertices.



The Crowding-Comparison Operator guides selection towards a uniformly spread-out (diverse) pareto-optimal front (best/first non-dominated front). Assume that now each member of the population (I) has two attributes:

1. Non-domination rank (*i*rank)
2. Crowding Distance (*i*distance)

This says prefer the solution with the better nondomination rank (nearer the pareto-optimal front) but if solutions are the same then prefer the one located in a less crowded region.

# C. Main Loop/Algorithm

1. Population is sorted using non-domination.
2. Each solution is assigned a fitness based equal to non-domination rank. (1 is the best)
3. Binary tournament selection, recombination & mutation operators applied (only for the initial generation)
4. The New population (Q) and the Previous One( P) are combined to make a large population (R) of double size (2N) which is then sorted according into non-dominated sets
5. The best sets are chosen to fill the slots in the next generation’s population (of normal size N). The best set f1 is put in then the next best set f2 and so on until a non-dominated set cannot fit in at which point the set is sorted by crowding distance and the remaining places are filled by those with the best (largest) crowding distance.
6. The new population (P+1) is used for selection, crossover and mutation to get (Q+1) and this procedure repeats from 4.