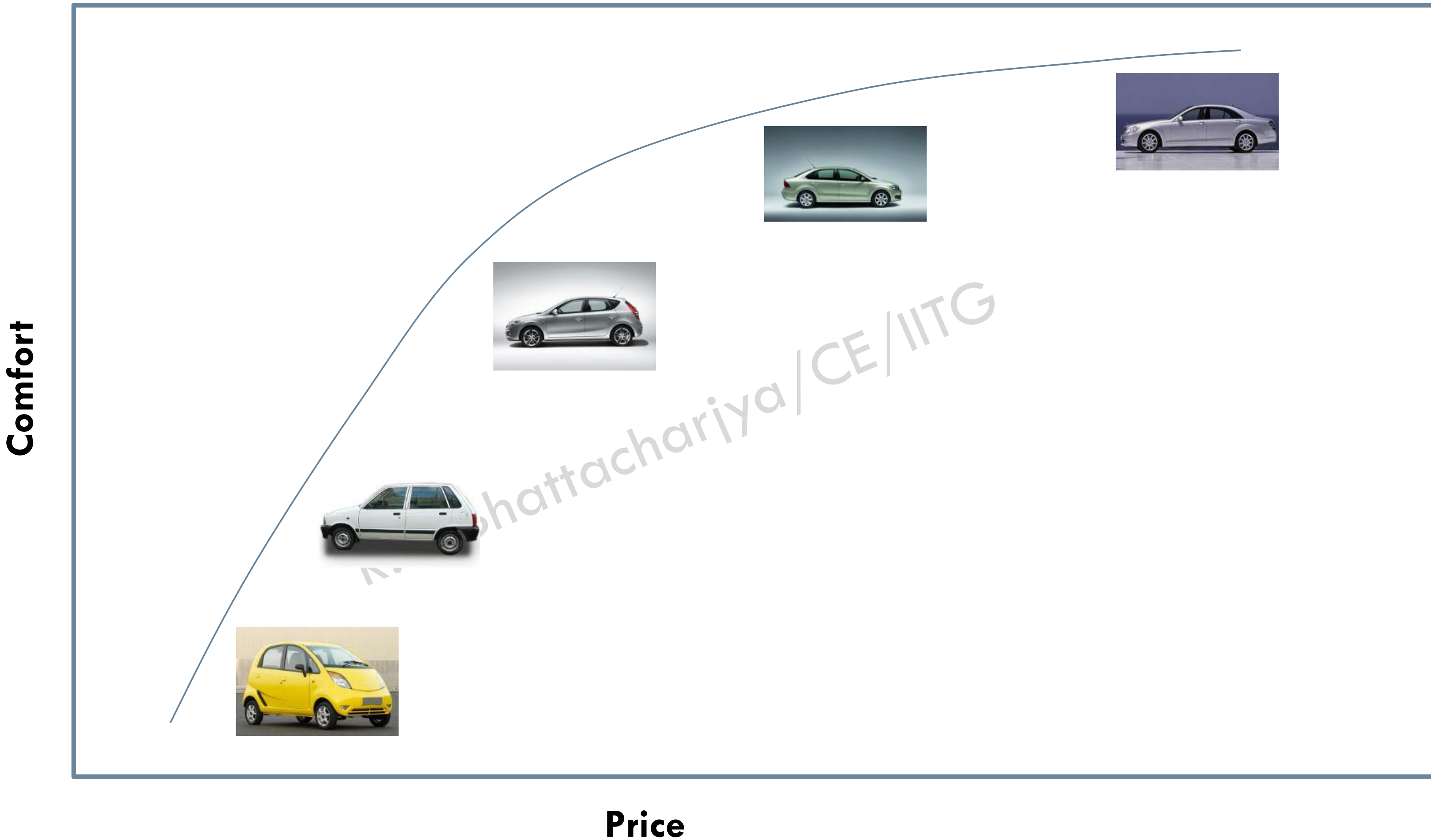
Multi-objective optimization

Dr. Rajib Kumar Bhattacharjya
Department of Civil Engineering
IIT Guwahati
Email: rkbc@iitg.ernet.in

6 November 2015

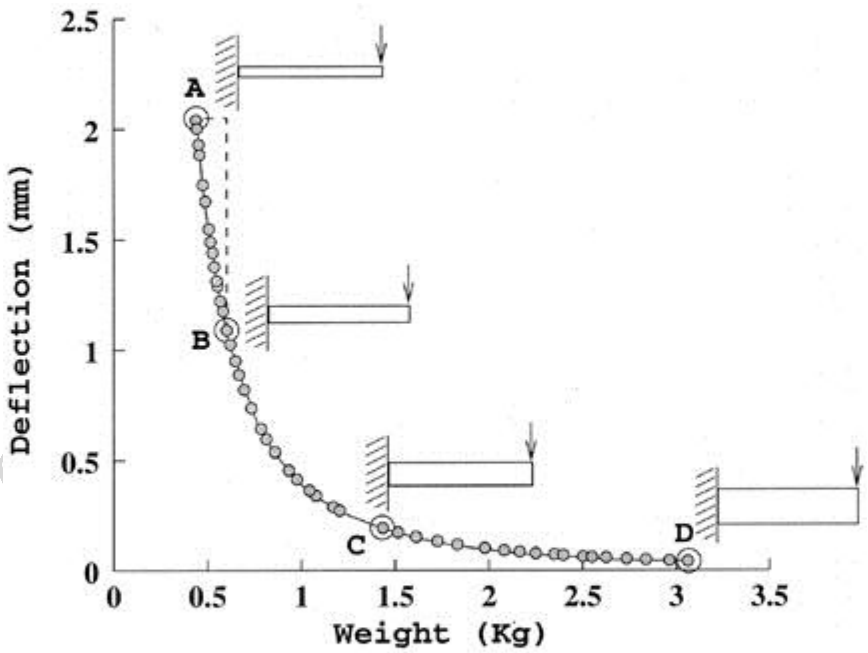
Multi-objective optimization



Multi-objective optimization

Two objectives are

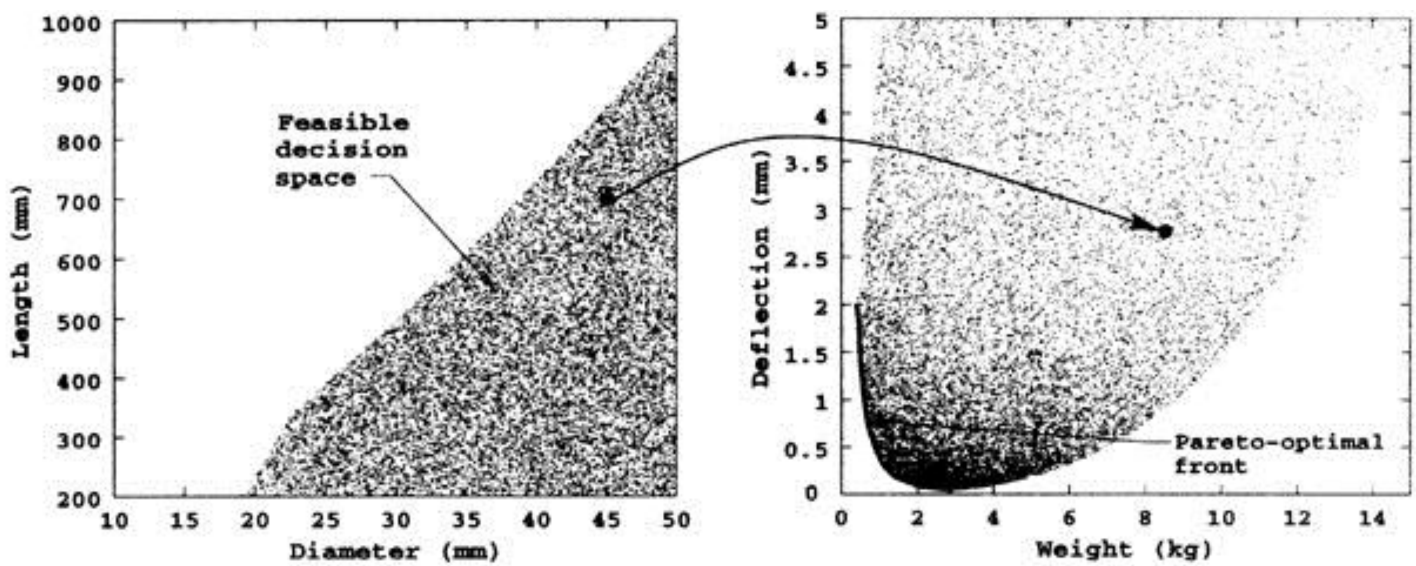
- Minimize weight
- Minimize deflection



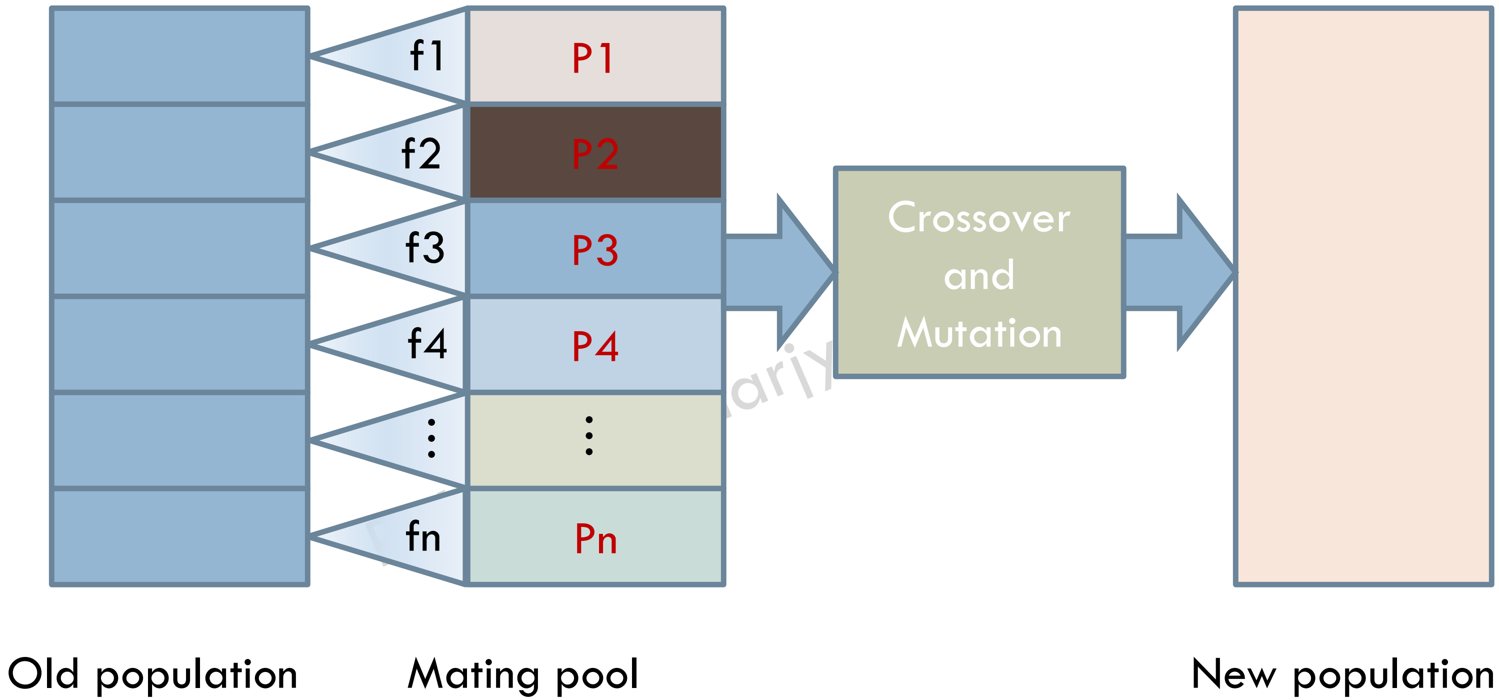
Multi-objective optimization

- More than one objectives
- Objectives are conflicting in nature
- Dealing with two search space
  - Decision variable space
  - Objective space
- Unique mapping between the objectives and often the mapping is non-linear
- Properties of the two search space are not similar
- Proximity of two solutions in one search space does not mean a proximity in other search space

Multi-objective optimization



Vector Evaluated Genetic Algorithm (VEGA)



Propose by Schaffer (1984)

Non-dominated selection heuristic

Give more emphasize on the non-dominated solutions of the population

This can be implemented by subtracting ∈ from the dominated solution fitness value

Suppose N' is the number of sub-population and n' is the non-dominated solutions. Then total reduction is $(N' - n')$ ∈.

The total reduction is then redistributed among the non-dominated solution by adding an amount $(N' - n')$ ∈$/n'$

This method has two main implications

Non-dominated solutions are given more importance

Additional equal emphasis has been given to all the non-dominated solution

Weighted based genetic algorithm (WBGA)

- The fitness is calculated

$$F = \sum_{j=1}^{M} w_j \frac{f_i - f_j^{min}}{f_j^{max} - f_j^{min}}$$

- The spread is maintained using the sharing function approach

Sharing function $$Sh(d_{ij}) = \begin{cases} 1 - (d_{ij}/\sigma), & \text{if } d_{ij} < \sigma\,; \\ 0, & \text{otherwise.} \end{cases}$$

Niche count $$nc_i = \sum_{j=1}^{N} Sh(d_{ij})$$

Modified fitness $$F' = \frac{F}{nc}$$

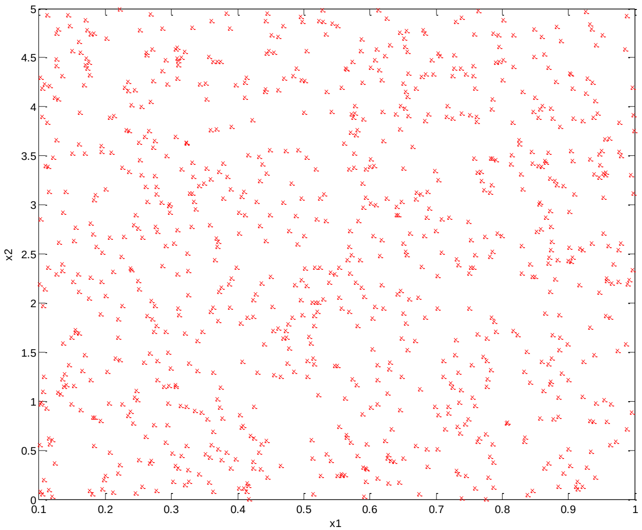
Multiple objective genetic algorithm (MOGA)

$$Maximize\ f_1 = 1.1 - x_1$$

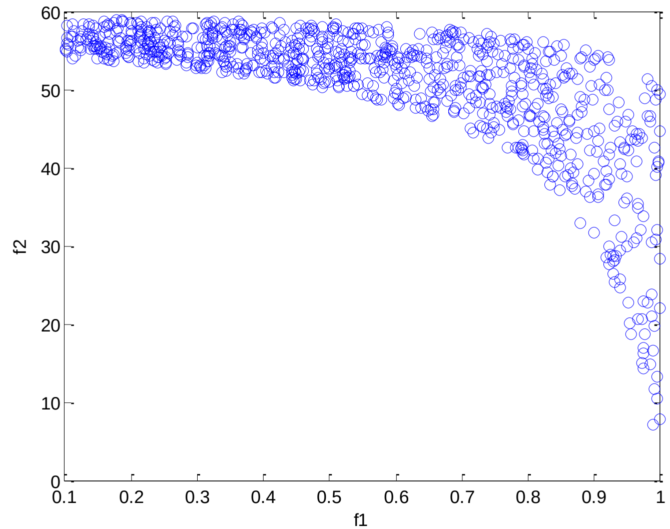
$$Maximize\ f_2 = 60 - \frac{1 + x_1}{x_2}$$

$$Subject\ to \quad 0.1 \le x_1 \le 1$$

$$0 \le x_2 \le 5$$

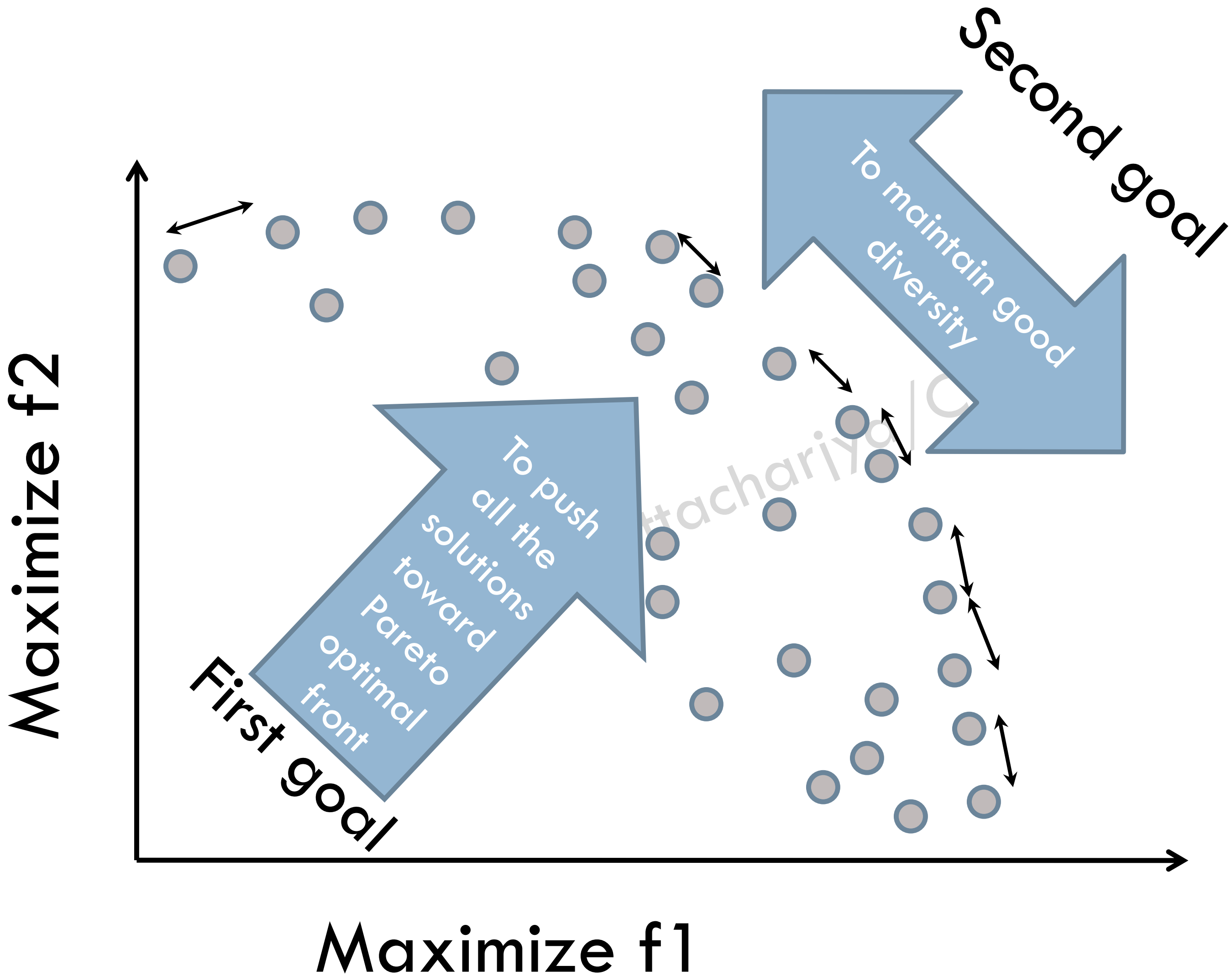


Solution space



Objective space

Multiple objective genetic algorithm (MOGA)

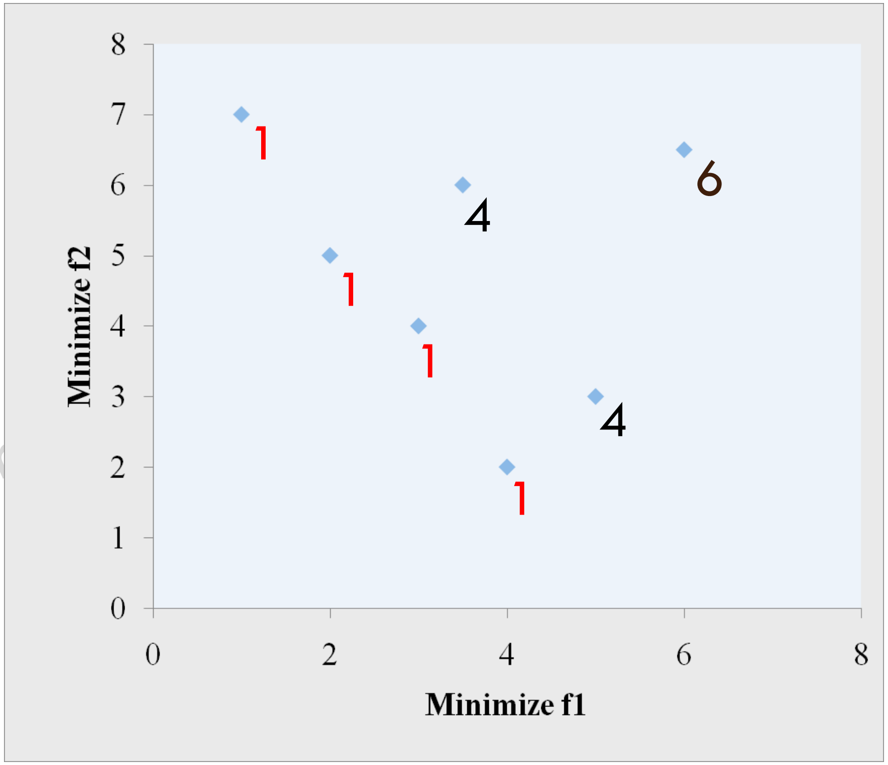


Multiple objective genetic algorithm (MOGA)

Fonseca and Fleming (1993) first introduced multiple objective genetic algorithm (MOGA)

The assigned fitness value based on the non-dominated ranking.

The rank is assigned as $r_i = 1 + n_i$ where r_i is the ranking of the i^{th} solution and n_i is the number of solutions that dominate the solution.



Multiple objective genetic algorithm (MOGA)

- Fonseca and Fleming (1993) maintain the diversity among the non-dominated solution using niching among the solution of same rank.
- The normalize distance was calculated as,

$$d_{i,j} = \sqrt{\sum_{k=1}^{M} \left(\frac{f_k^i - f_k^j}{f_k^{max} - f_k^{min}} \right)^2}$$

- The niche count was calculated as,

$$nc_i = \sum_{j=1}^{\mu(r_i)} Sh(d_{ij})$$

NSGA

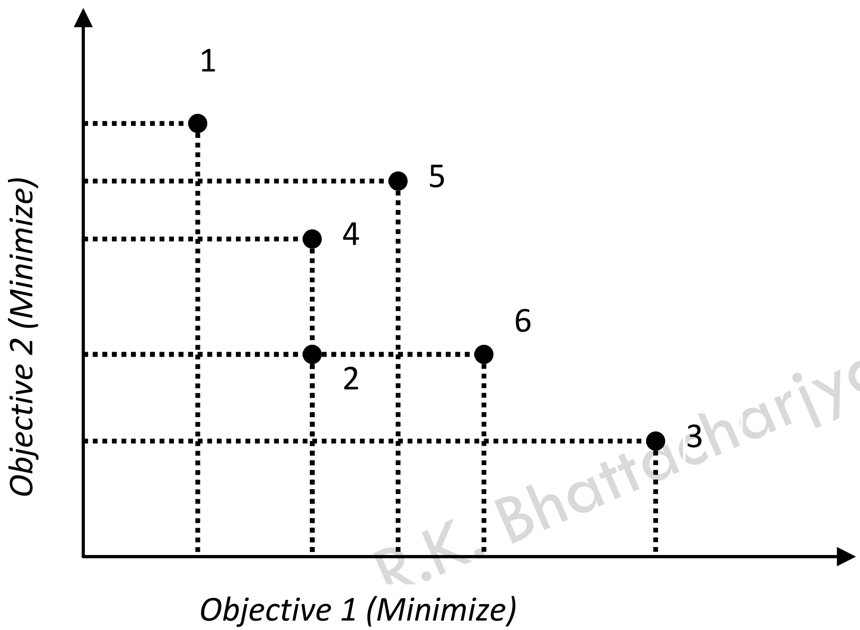
- Srinivas and Deb (1994) proposed NSGA
- The algorithm is based on the non-dominated sorting.
- The spread on the Pareto optimal front is maintained using sharing function

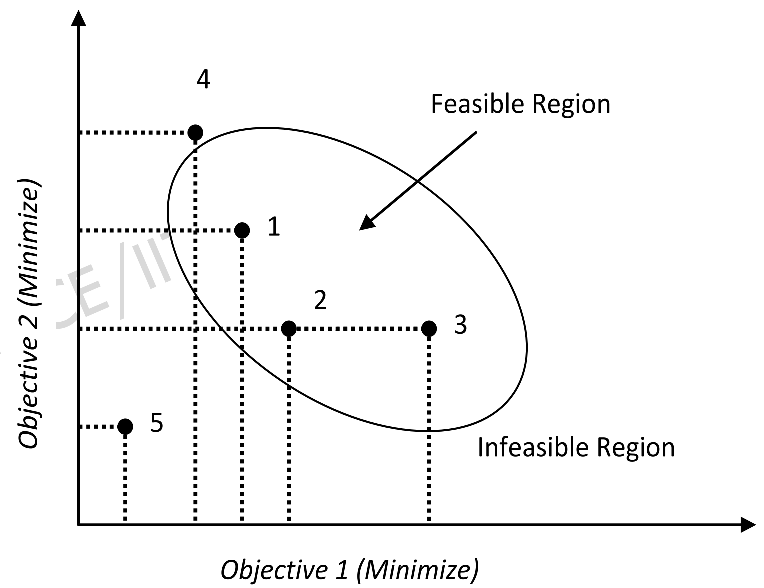
$$d_{i,j} = \sqrt{\sum_{k=1}^{P_1} \left(\frac{x_k^i - x_k^j}{x_k^{max} - x_k^{min}} \right)^2}$$

NSGA II

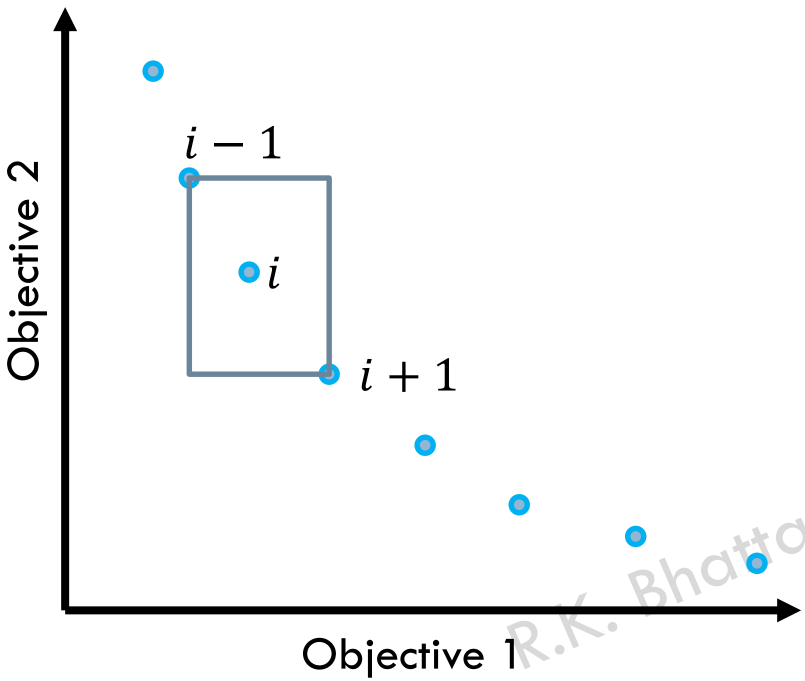
- Non-dominated Sorting Genetic Algorithms
  - NSGA II is an elitist non-dominated sorting Genetic Algorithm to solve multi-objective optimization problem developed by Prof. K. Deb and his student at IIT Kanpur.
  - It has been reported that NSGA II can converge to the global Pareto-optimal front and can maintain the diversity of population on the Pareto-optimal front

Non-dominated sorting





Calculation crowding distance



Cd, the crowded distance is the perimeter of the rectangle constituted by the two neighboring solutions

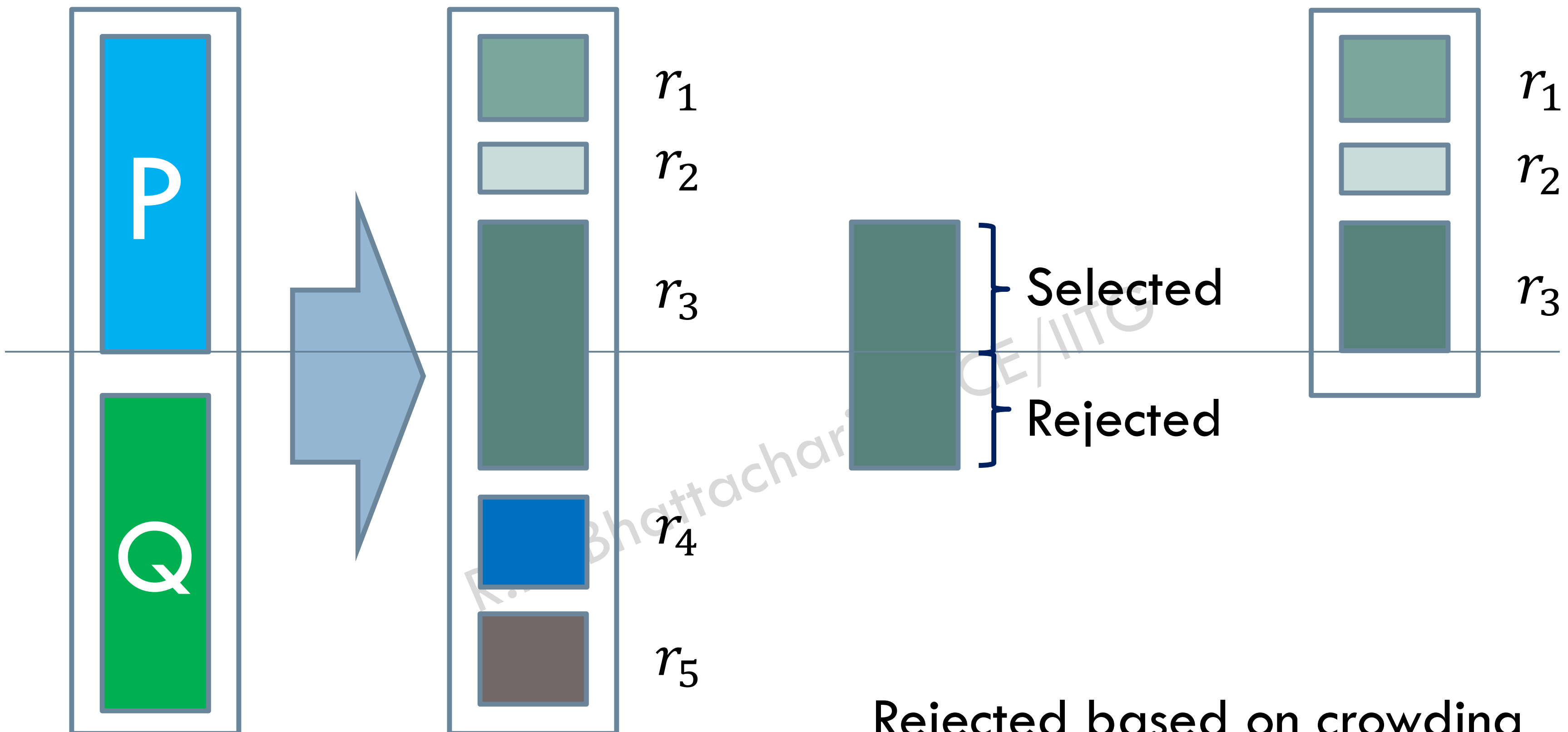
Cd value more means that the solution is less crowded

Cd value less means that the solution is more crowded

Crowded tournament operator

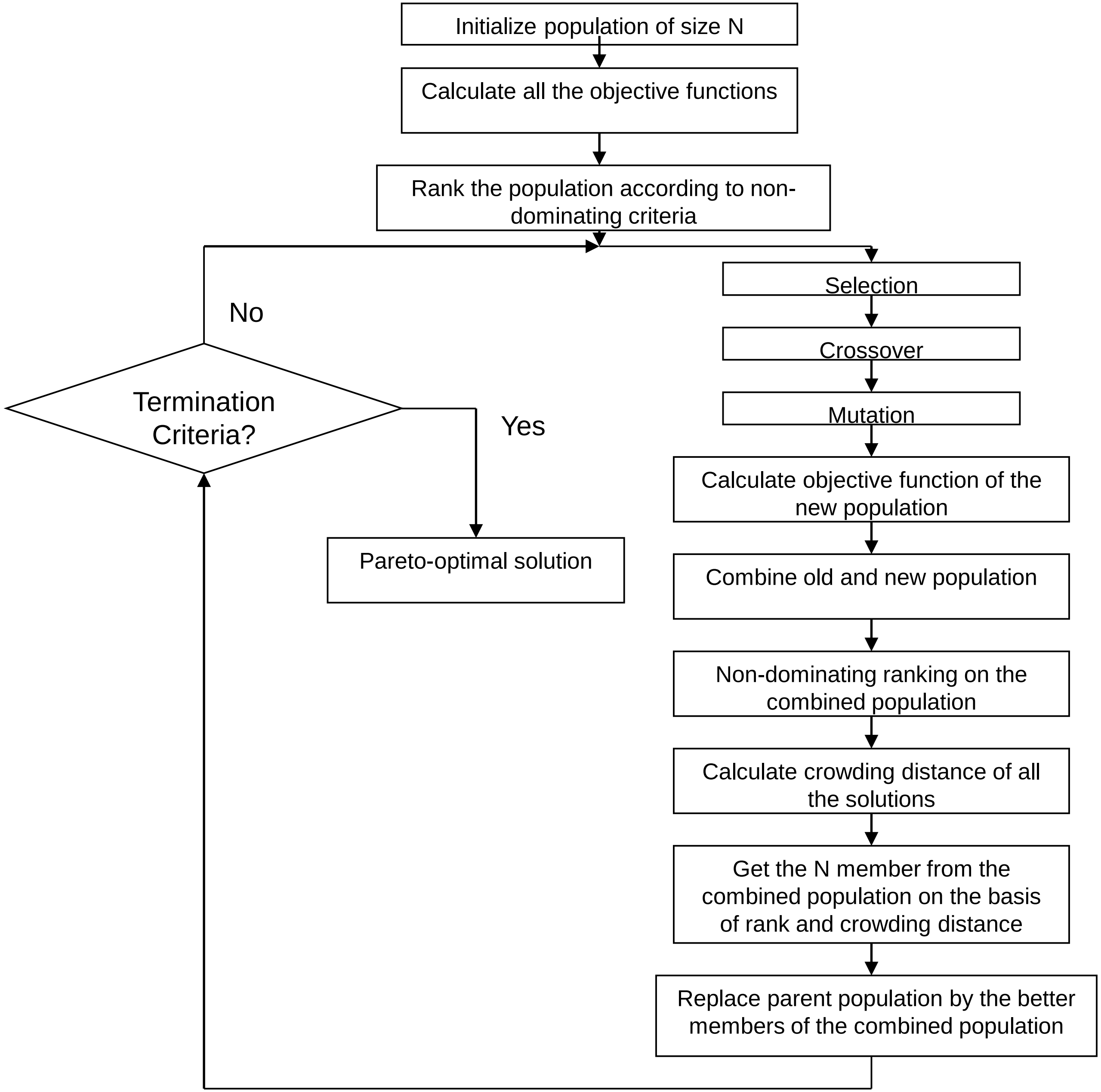
- A solution i wins a tournament with another solution j,
  - If the solution i has better rank than j, i.e. $r_i < r_j$
  - If they have the same rank, but i has a better crowding distance than j, i.e. $r_i = r_j$ and $d_i > d_j$.

Replacement scheme of NSGA II



Non dominated sorting

Rejected based on crowding distance


Initialize population of size N
Calculate all the objective functions
Rank the population according to non-dominating criteria
Selection
Crossover
Mutation
Calculate objective function of the new population
Combine old and new population
Non-dominating ranking on the combined population
Calculate crowding distance of all the solutions
Get the N member from the combined population on the basis of rank and crowding distance
Replace parent population by the better members of the combined population
Termination Criteria?
No
Yes
Pareto-optimal solution

THANKS