Bernabé Dorronsoro
Patricia Ruiz

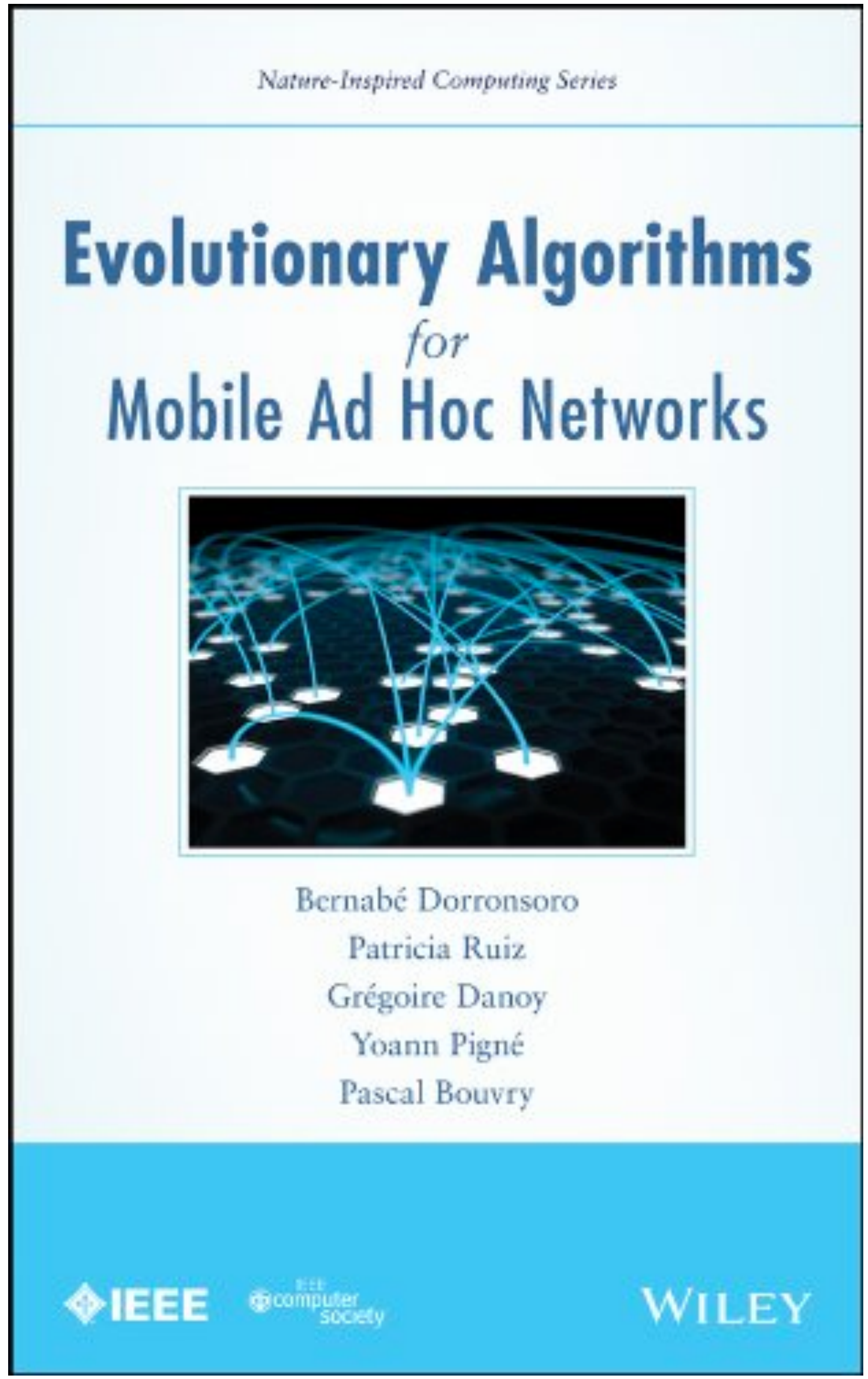
bernabe.dorronsoro@uca.es
www.bernabe.dorronsoro.es
patricia.ruiz@uca.es
patriciaruiz.es

Universidad de Cádiz

An Optimization Framework for Mobile Ad Hoc Networks

On the Need of Optimization for Mobile Ad Hoc Networks

Book


Nature-Inspired Computing Series
Evolutionary Algorithms
for
Mobile Ad Hoc Networks
Bernabé Dorronsoro
Patricia Ruiz
Grégoire Danoy
Yoann Pigné
Pascal Bouvry
IEEE
computer society
WILEY

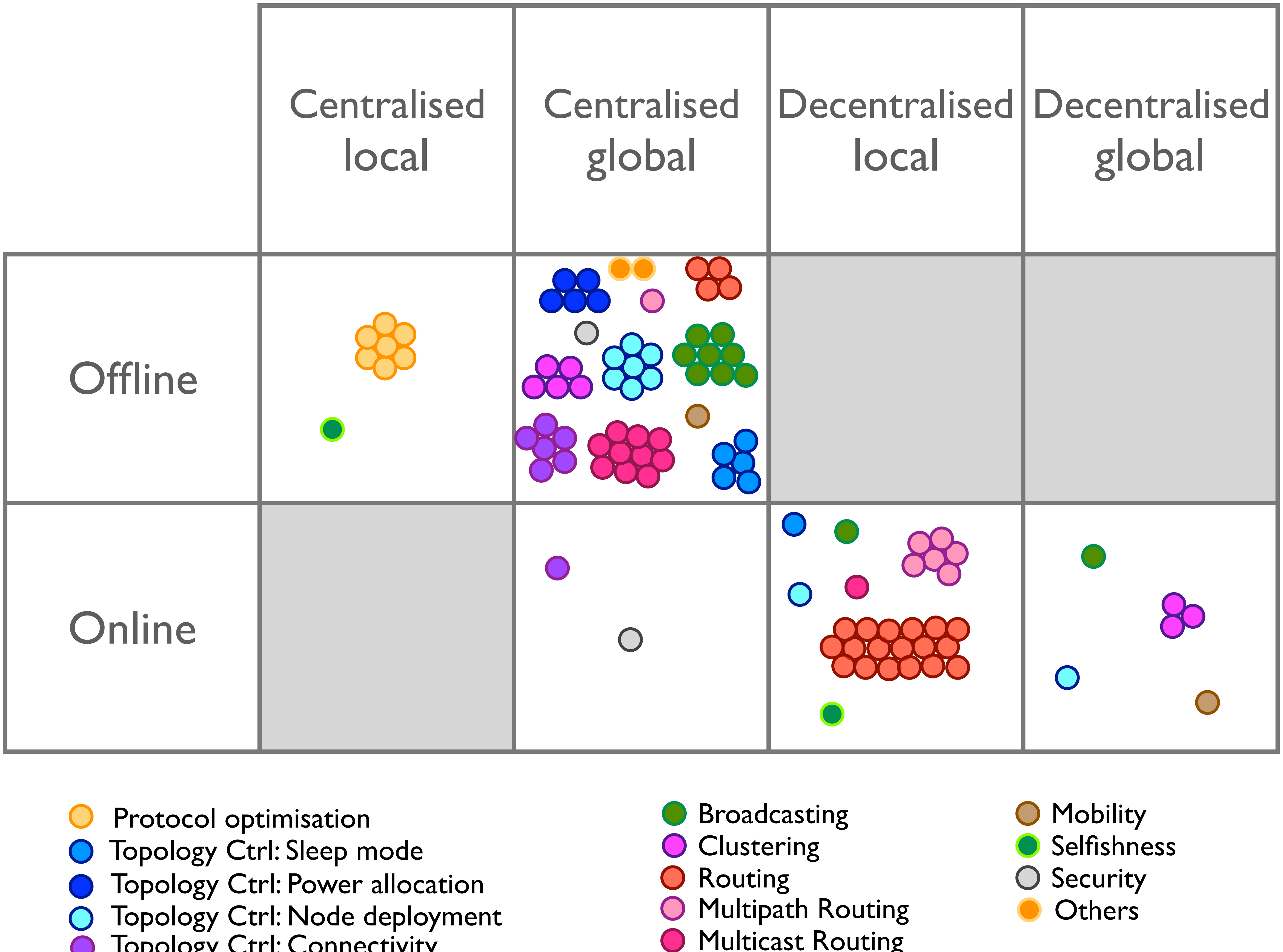
Metaheuristics and MANETs

- Energy efficiency
- Broadcast
- Routing
- Network topology
  - Connectivity
  - Clustering
  - Node deployment
- Selfishness
- Security
- Quality of Service

Metaheuristics and MANETs

- We can characterise them in terms of
  - Operation mode
    - Offline
    - Online
  - Knowledge
    - Global
    - Local
  - Approach
    - Centralized
    - Decentralized

Metaheuristics and MANETs

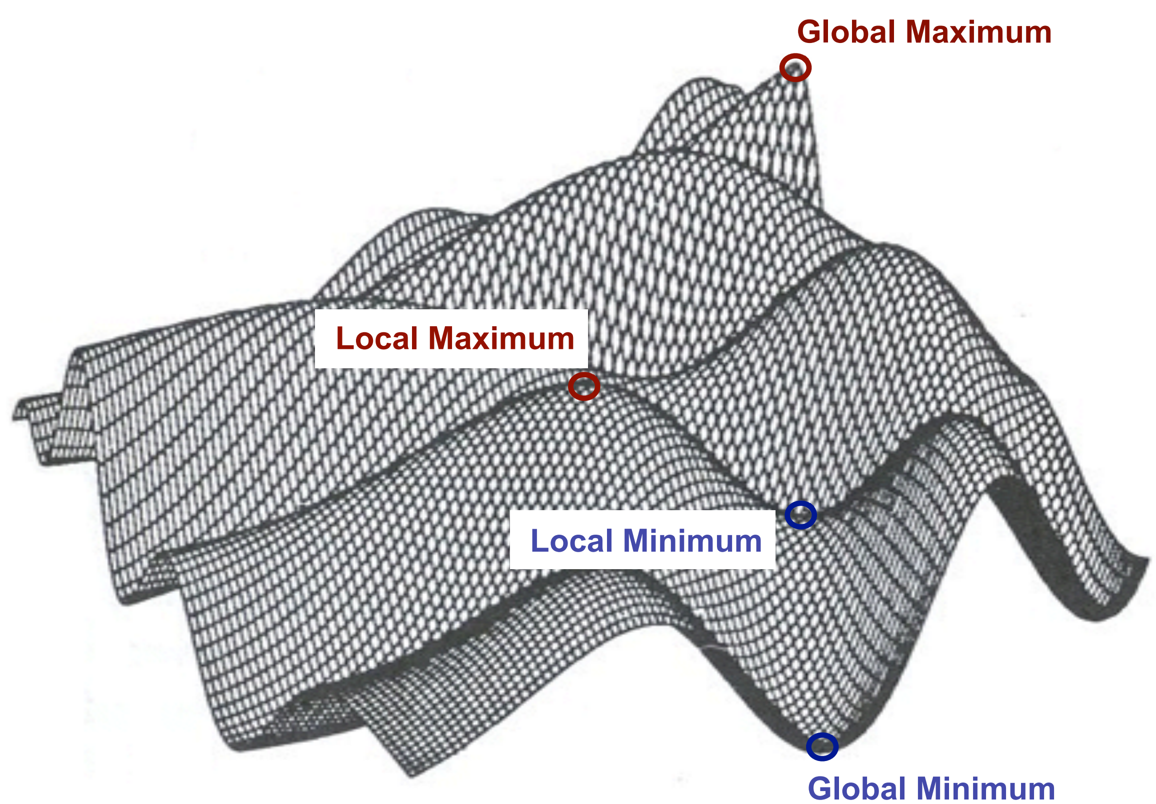


Single-objective Optimization

Single-objective optimization

Solutions Space

$$Opt(f(\vec{x})) = \{\vec{x}^* \mid \forall\, \vec{x} \in M : \begin{cases} f(\vec{x}) \leq f(\vec{x}^*)\} & \textbf{Maximization} \\ f(\vec{x}) \geq f(\vec{x}^*)\} & \textbf{Minimization} \end{cases}$$



How to solve optimization problems?

- Complete methods

They guarantee to find for every finite size instance of a CO problem an optimal solution in bounded time

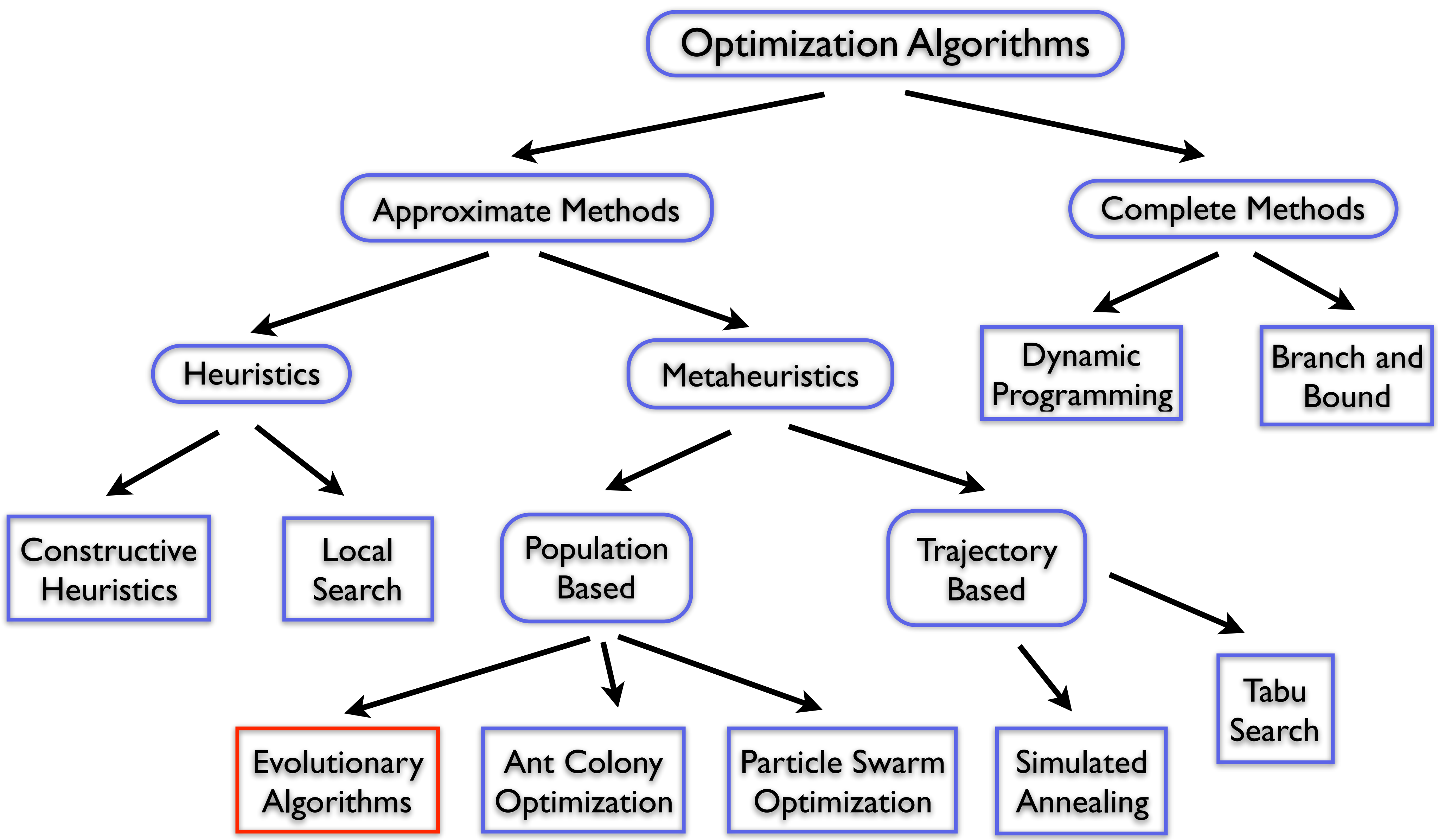
Only CO problems!

- Approximate methods

No guarantee of finding an optimal solution

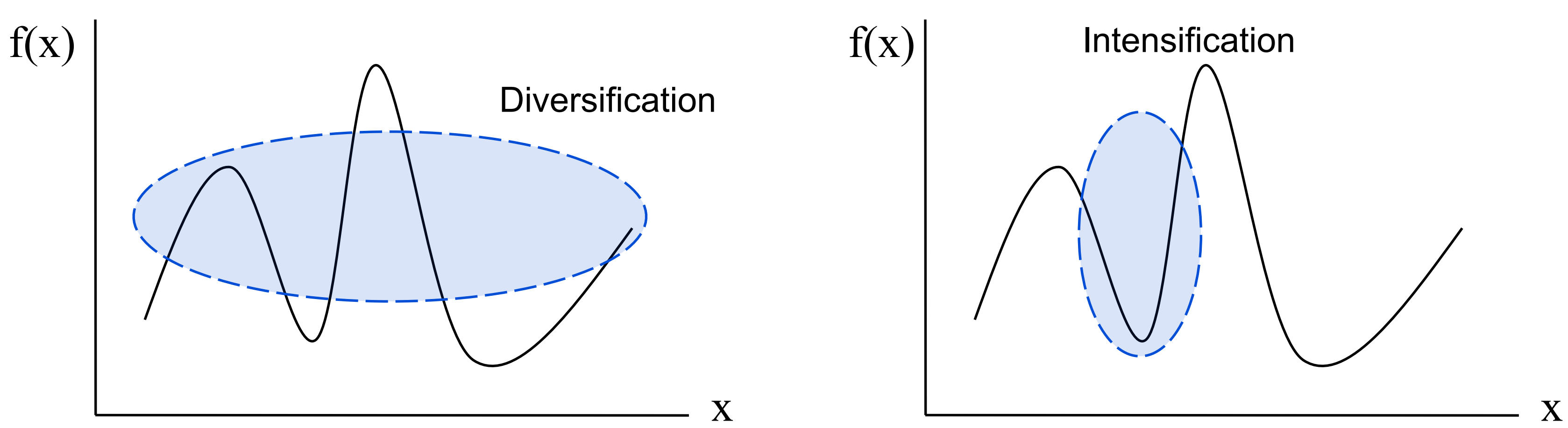
Combinatorial and Continuous

Optimization algorithms classification



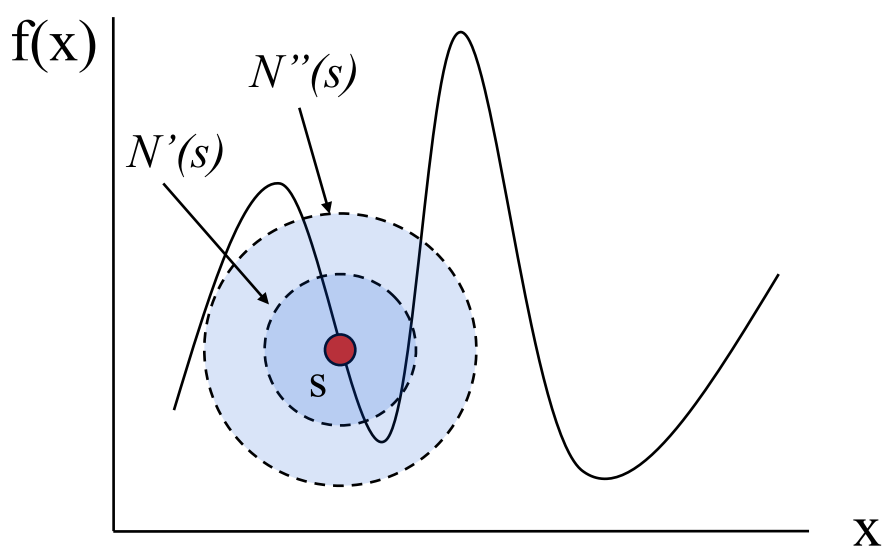
Diversification/Intensification

- Metaheuristics must achieve a balance between diversification and intensification
  - Diversification: exploration of the search space
  - Intensification: exploitation of promising regions of the search space



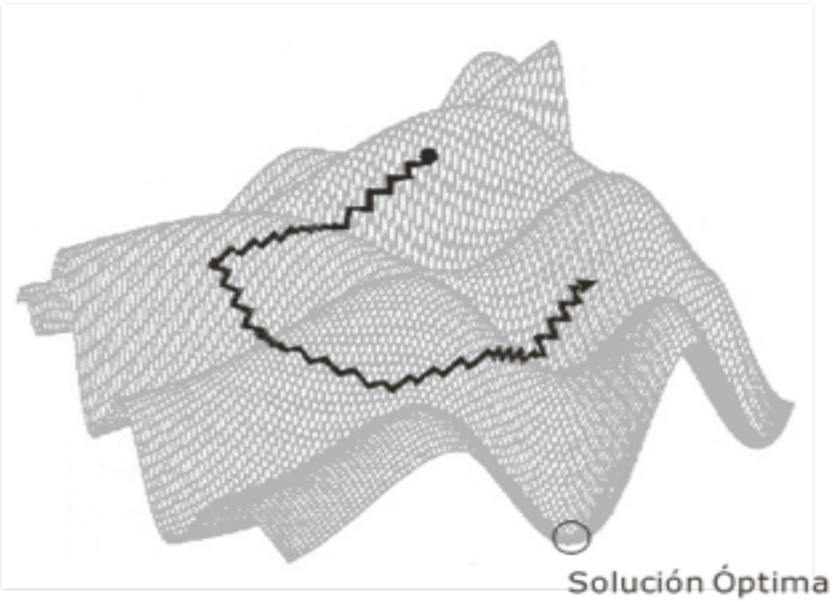
Concept of neighborhood

- Given a solution s, the neighborhood of s, N(s), is the set of solutions of the search space that can be reached using some kind of transformation on s



Genetic algorithms

- Evolutionary Algorithms (EAs): Useful optimization techniques for complex problems
  - Show a good tradeoff between exploration and exploitation
- Based in population
  - Individuals ➔ Potential solutions to the problem
    - Fitness value: ¿How good is the individual?
  - Variation operators ➔ Allow the evolution of the population
    - Recombination: Interchange of genetic material
    - Mutation: Generation of new genetic material

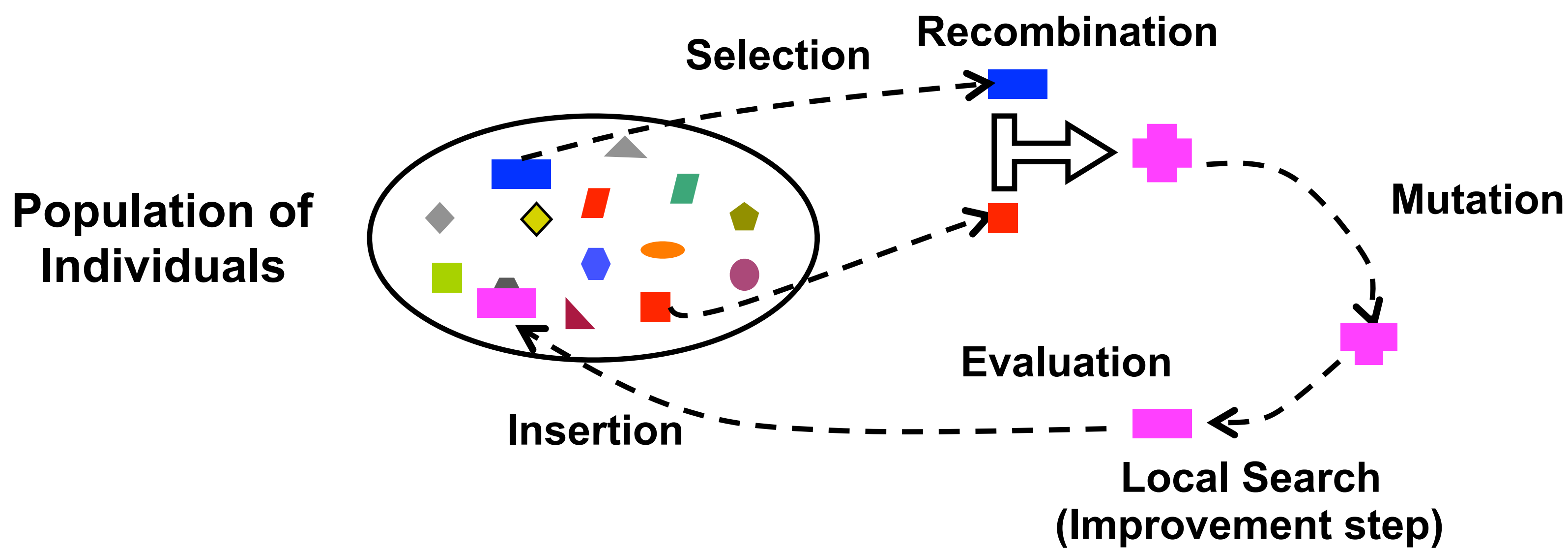


Genetic algorithms

- Population evolution
  - Improvement of the quality of solutions
  - Guided by the fitness function

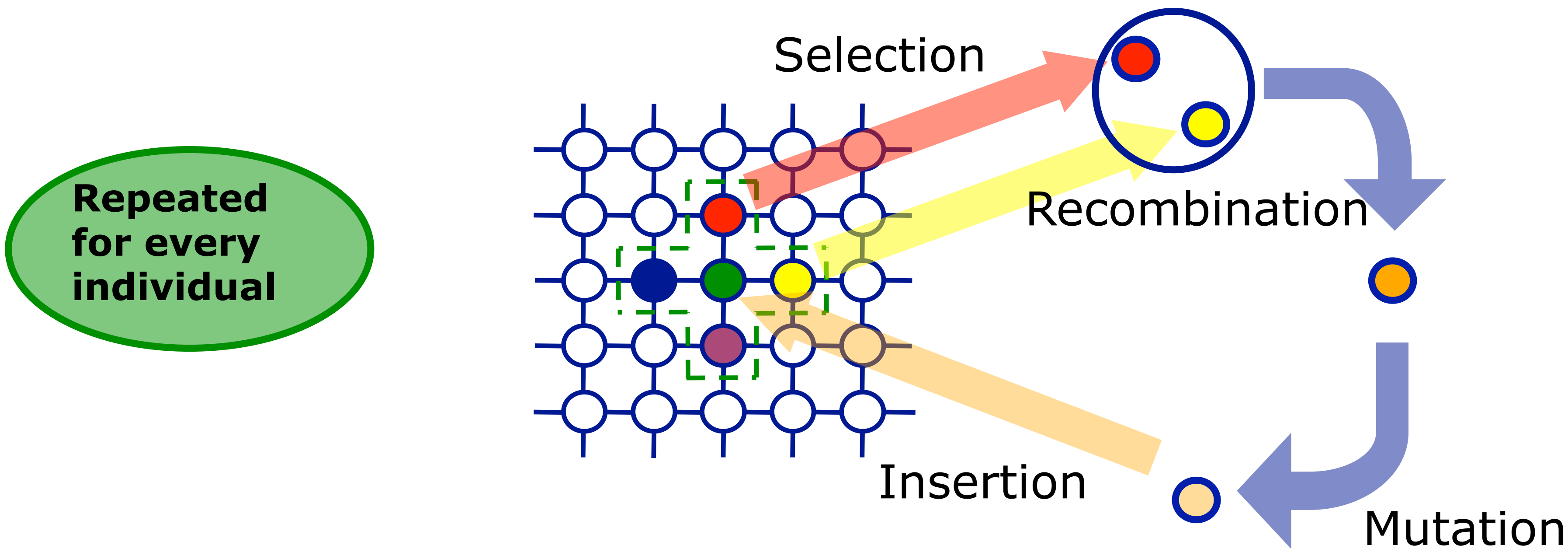
Panmictic

- Application operators
  - Stochastic
  - Generic



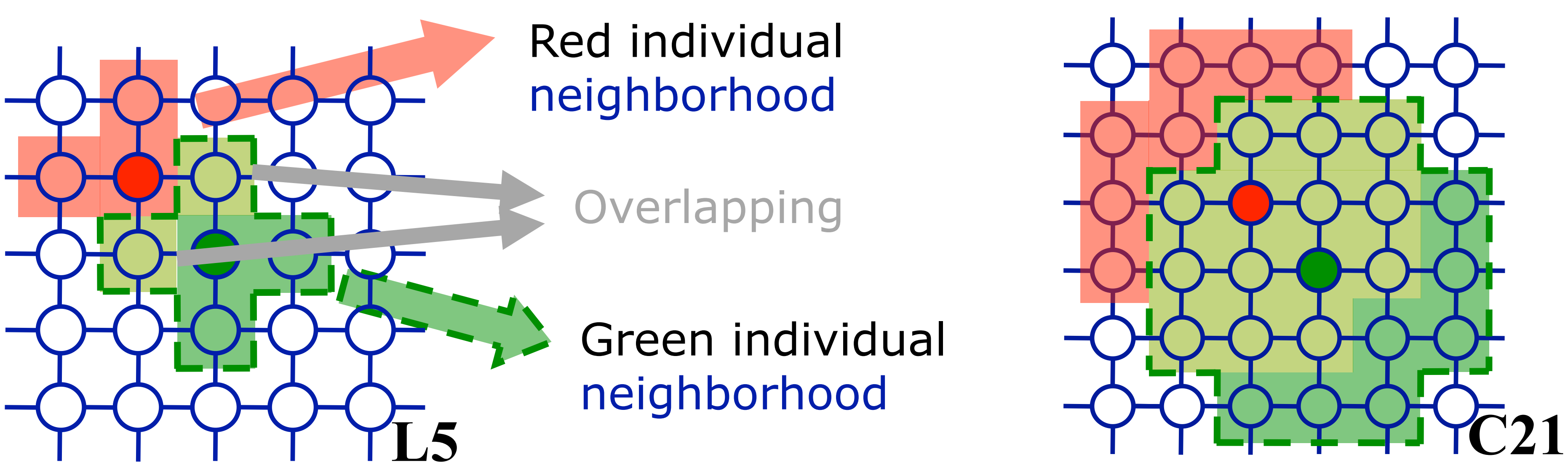
Cellular genetic algorithms

- Spatially structured population (2-D)
- Breeding loop applied inside small neighborhoods



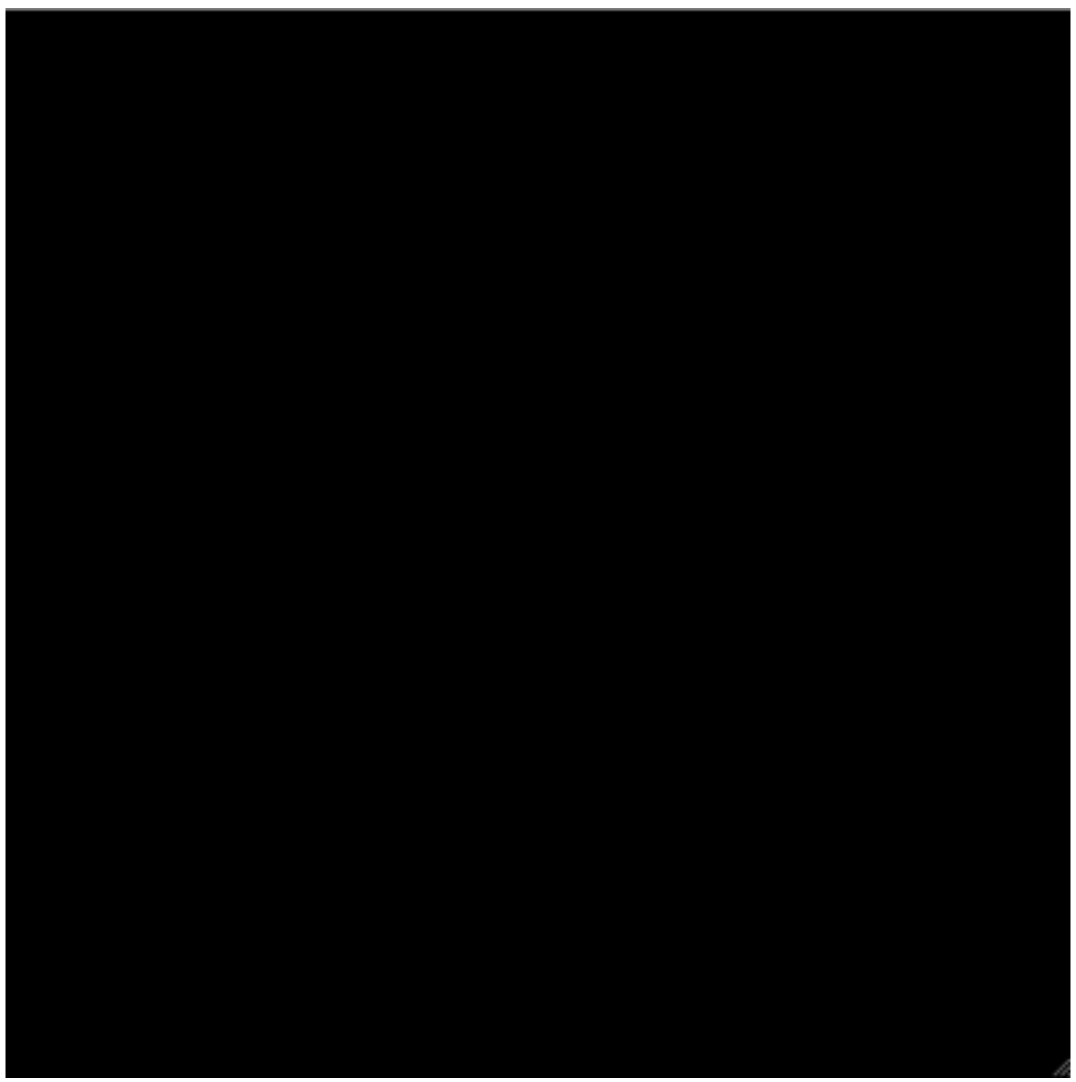
Cellular genetic algorithms

- Spatially structured population (2-D)
- Breeding loop applied inside small neighborhoods
- Overlapped neighborhoods ➔ Smooth diffusion
- Isolation by distance among individuals in the population
- Appropriate exploration/exploitation tradeoff
  - Exploitation: Inside neighborhoods
  - Exploration: Neighborhood borders



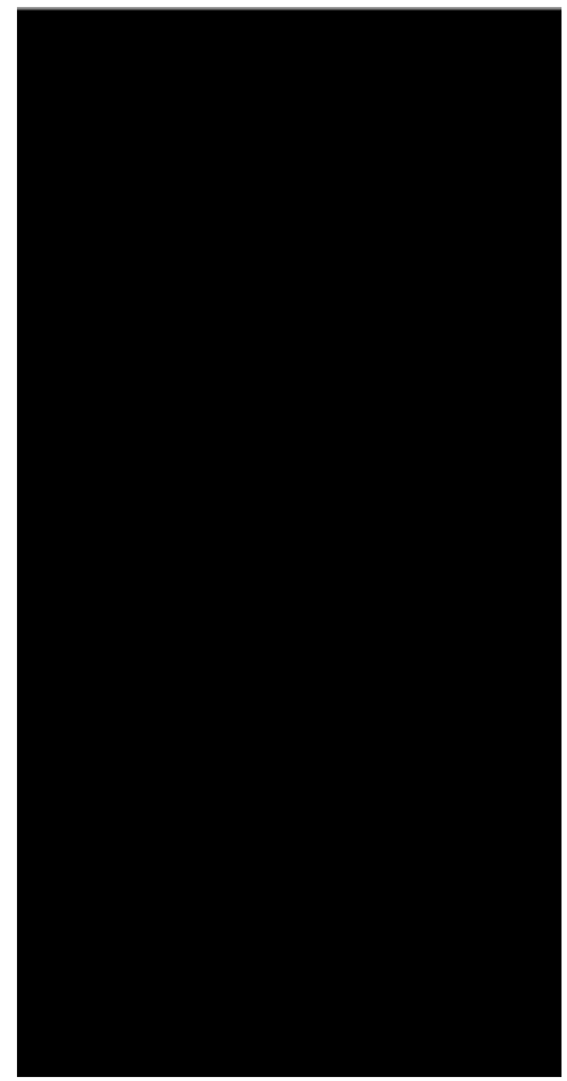
MAXUT100 Problem

cGA with L5

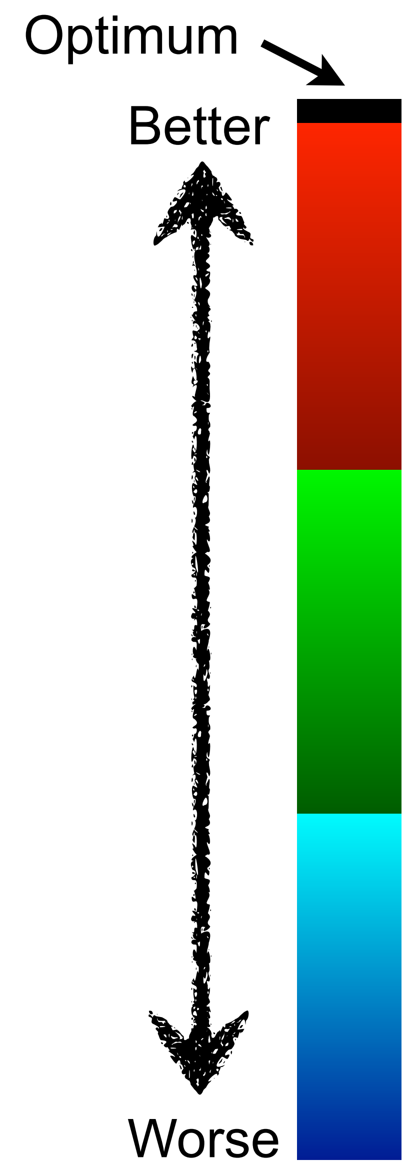


Optimum (1077.0) after 33 s

genGA



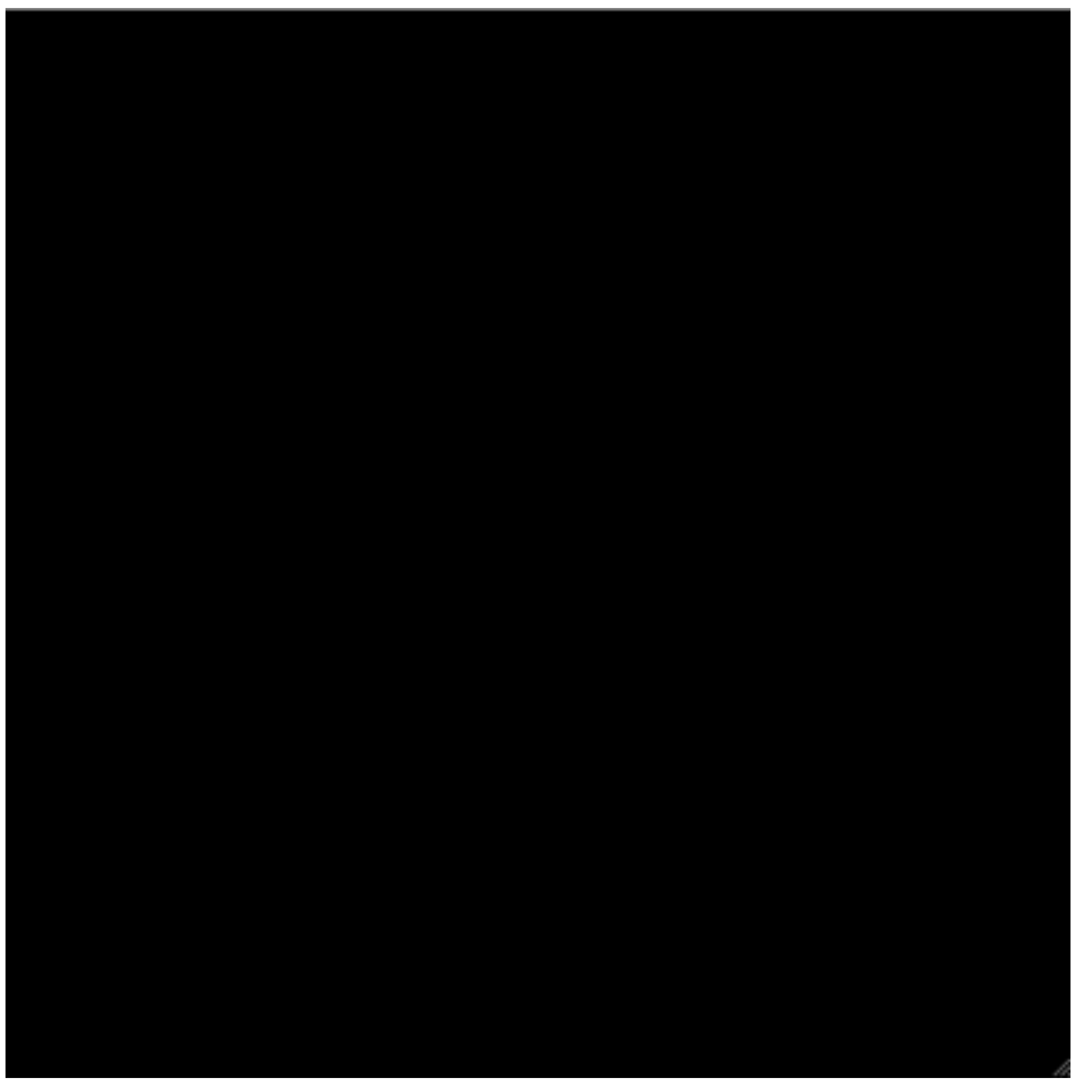
Converges to 967.0 after 24s



Cellular genetic algorithms

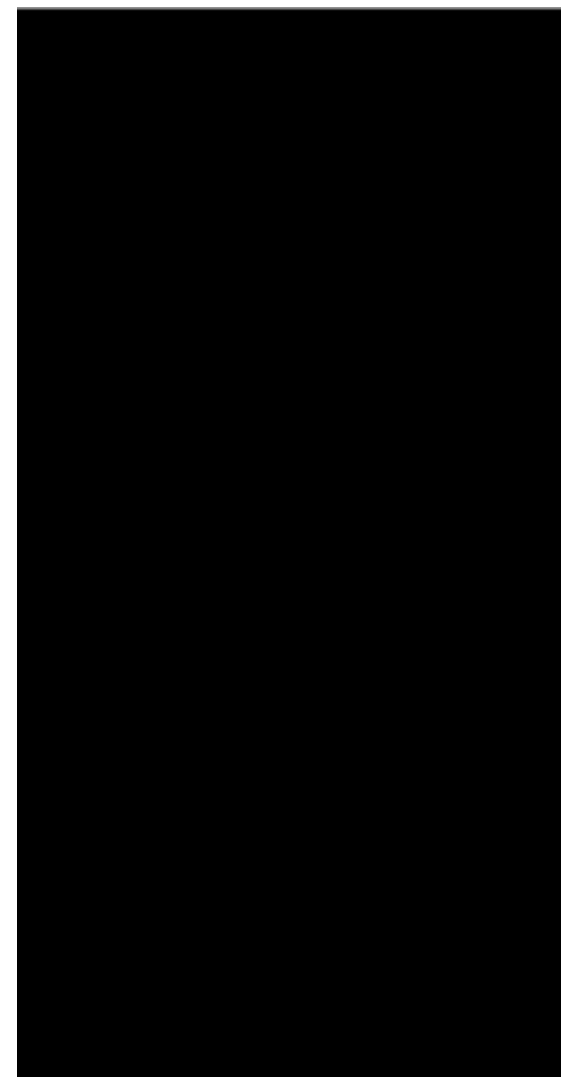
MAXUT100 Problem

cGA with L5

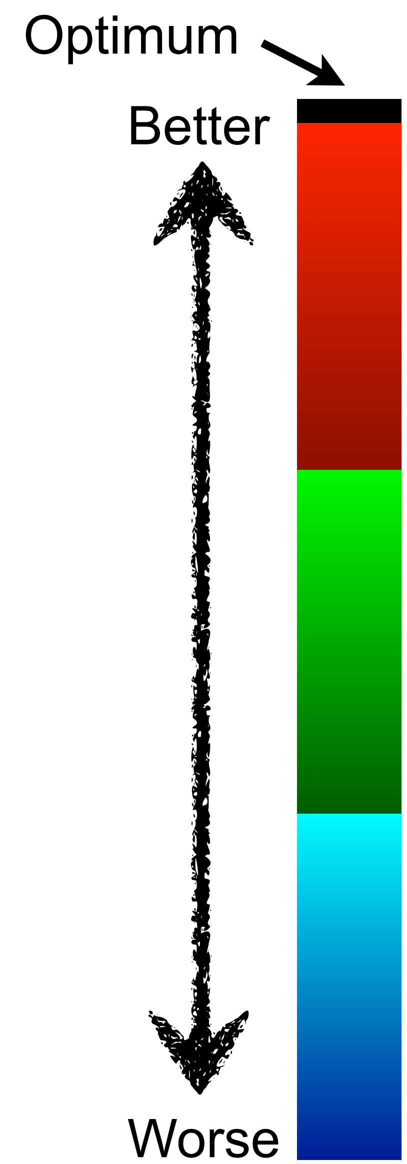


Optimum (1077.0) after 33 s

genGA



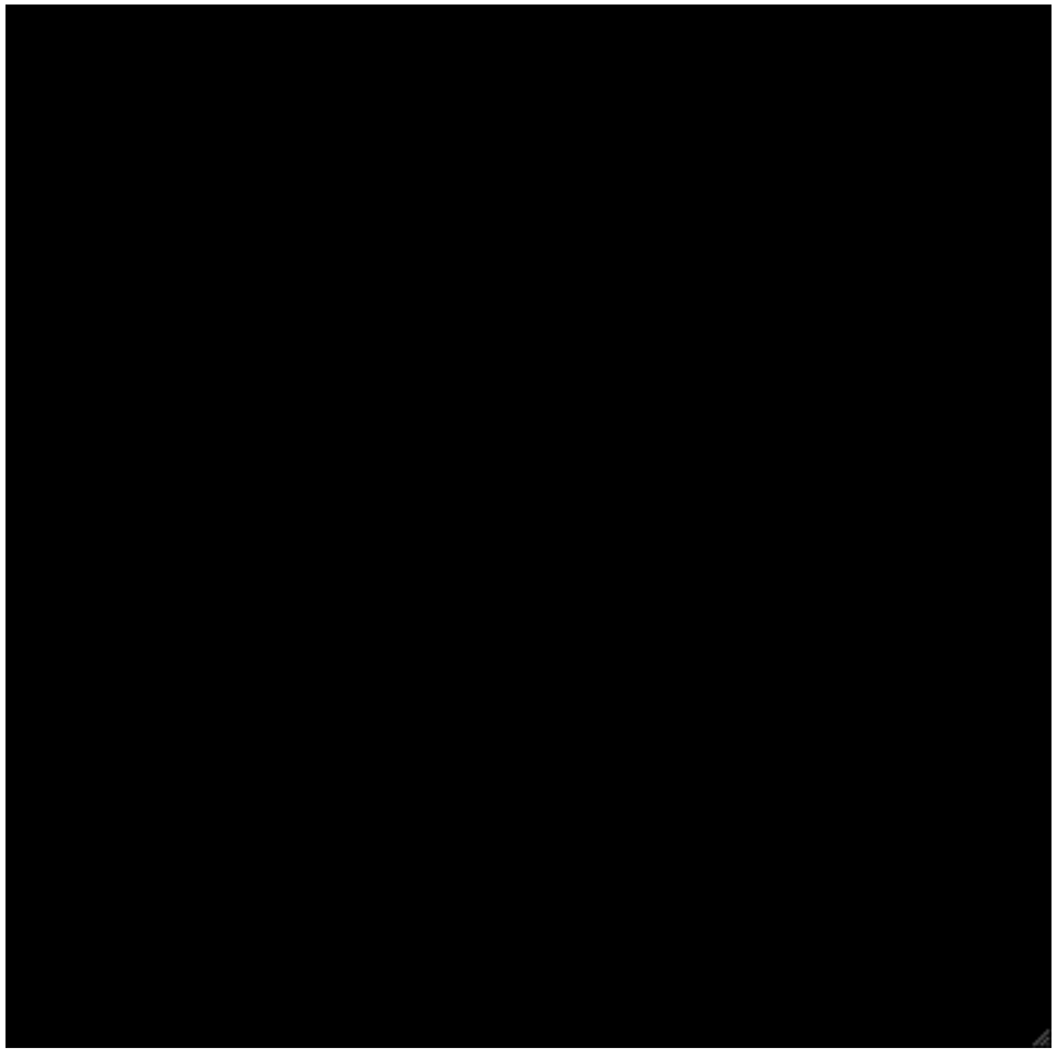
Converges to 967.0 after 24s



Cellular genetic algorithms

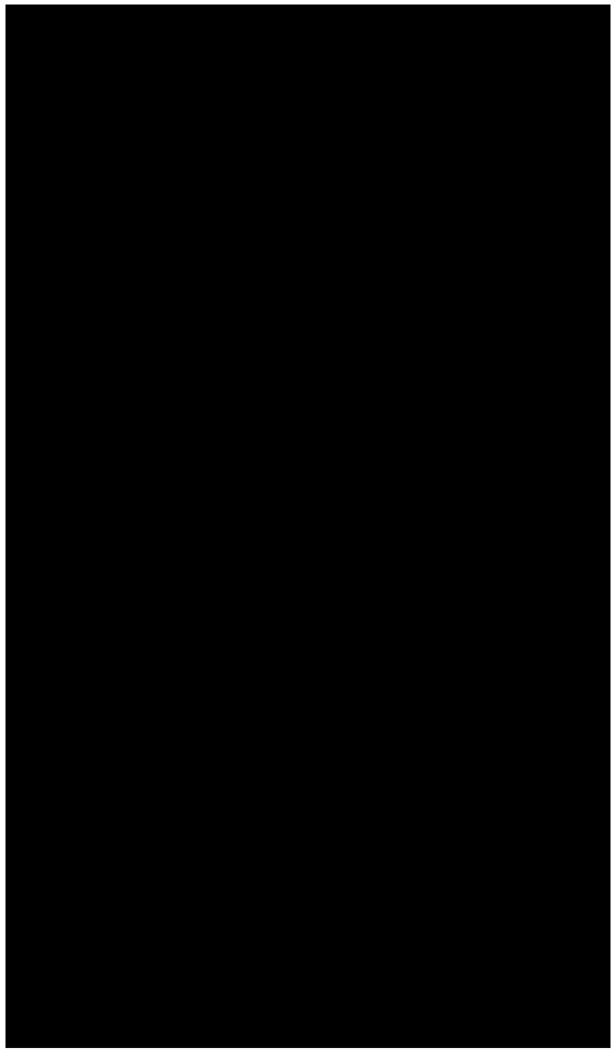
MAXUT20_01 Problem

cGA with L5

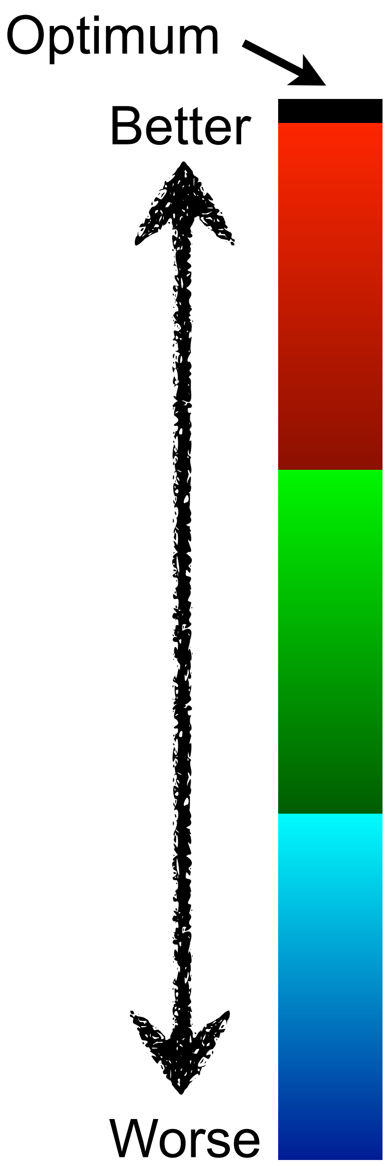


Optimum (10.1198) after 1.9 s

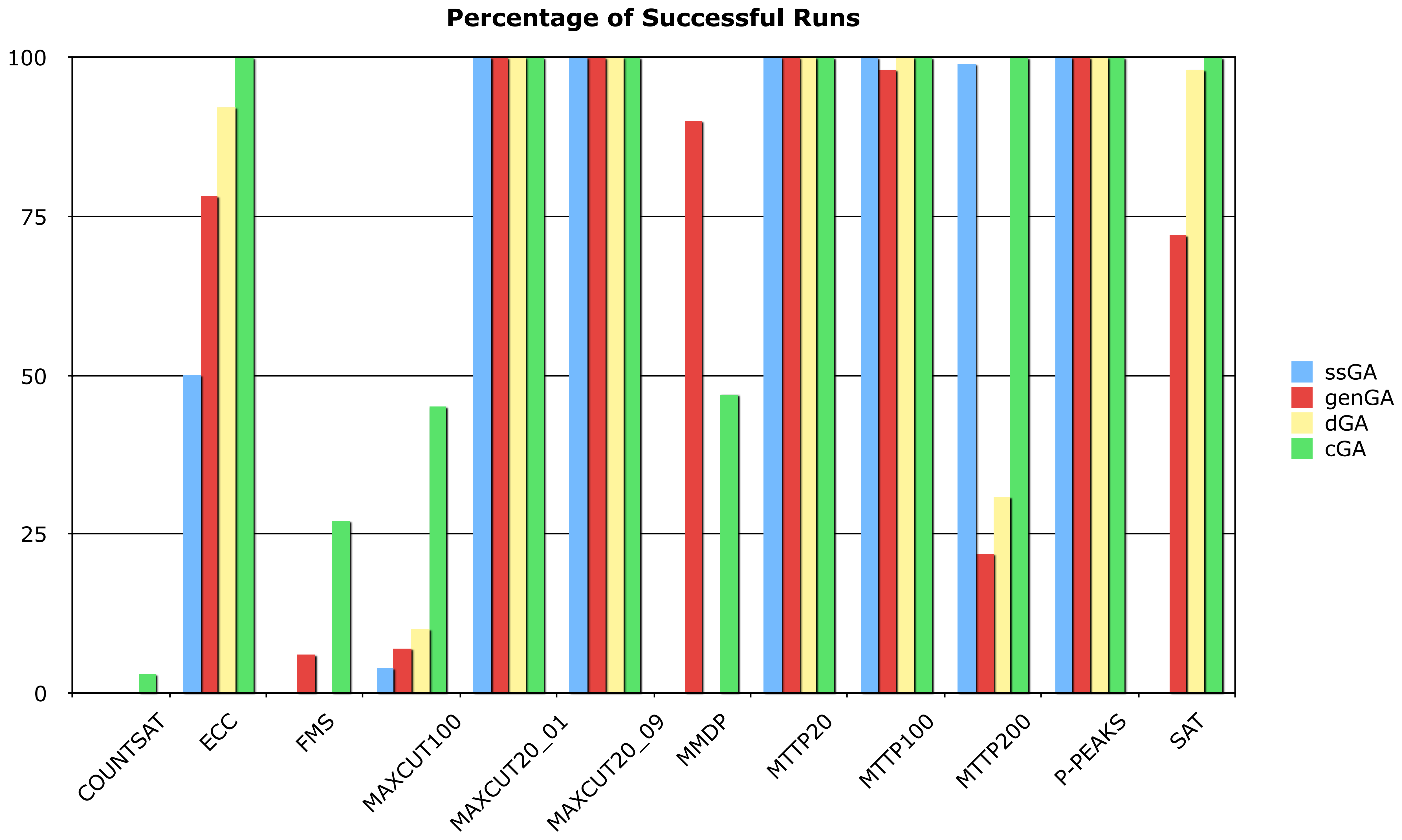
genGA



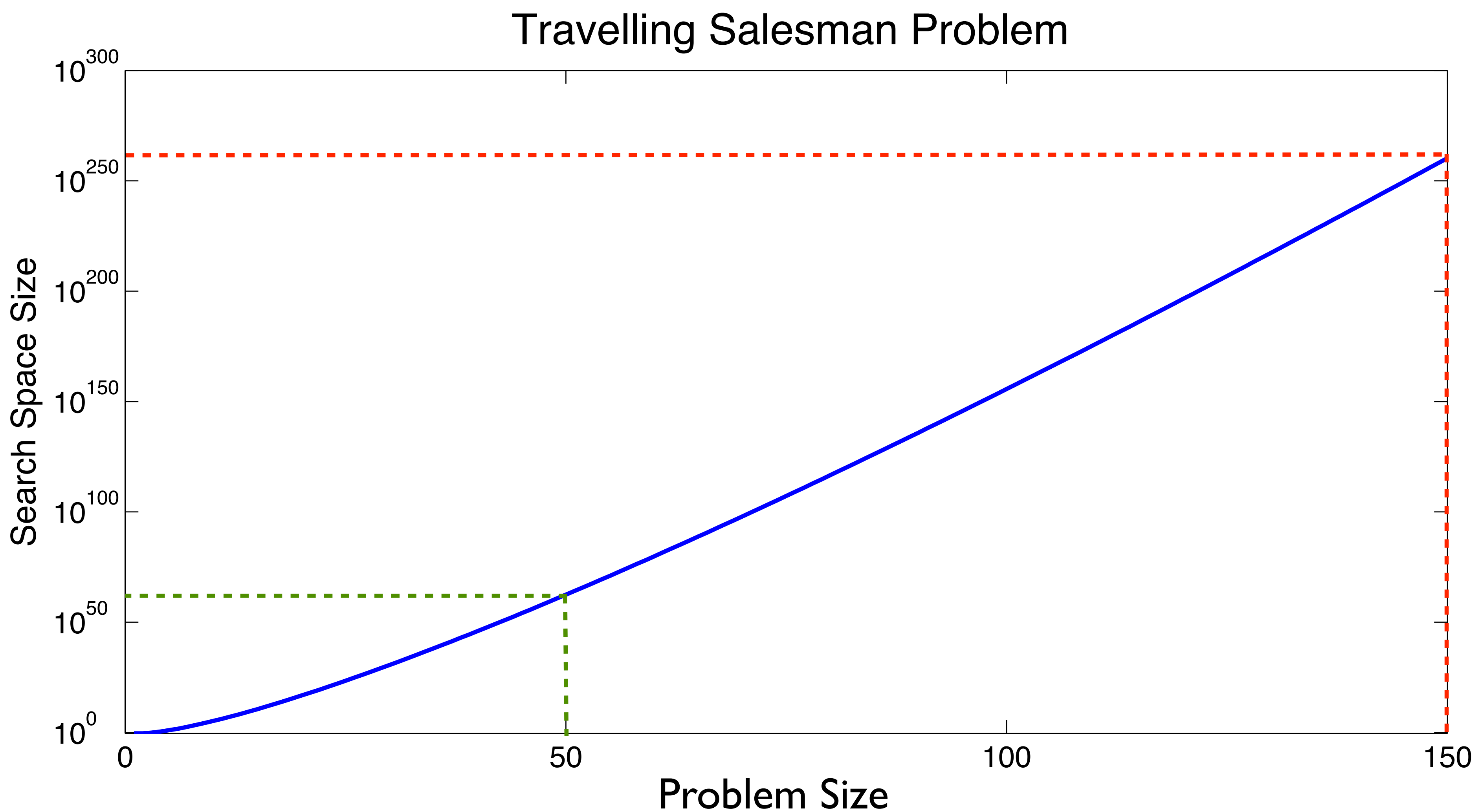
Optimum after 18.5s



Cellular genetic algorithms



Cooperative Coevolutionary GAs



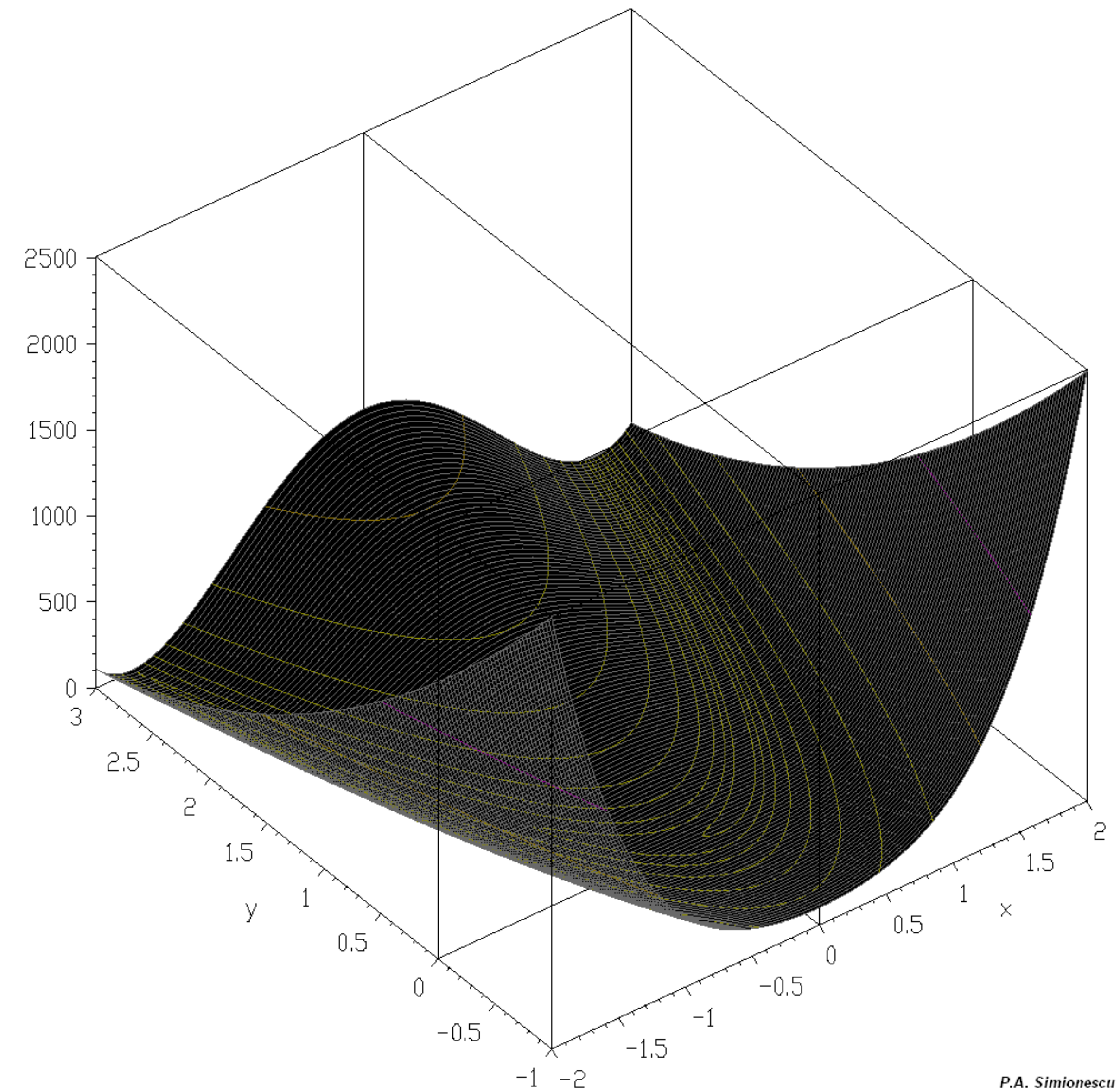
Rosenbrock function

- Part of De Jong's five function test suite
- Continuous and unimodal

$$f(x) = \sum_{i=1}^{n}\left(100\left(x_i^2 - x_{i+1}\right)^2 + \left(1 - x_i\right)^2\right)$$

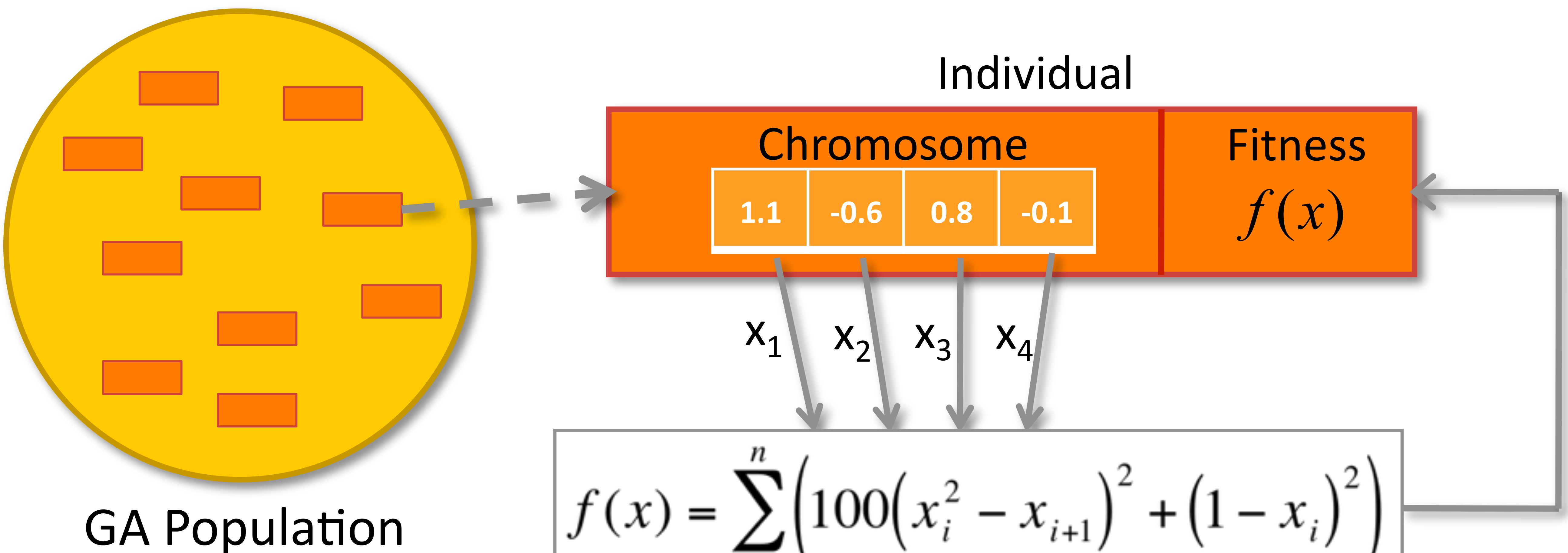
with -2.12 ≤ xi ≤ 2.12

- Global minimum $f(x^*) = 0$ with $x^* = (1,1,\ldots 1)$



GA on Rosenbrock (4 variables)

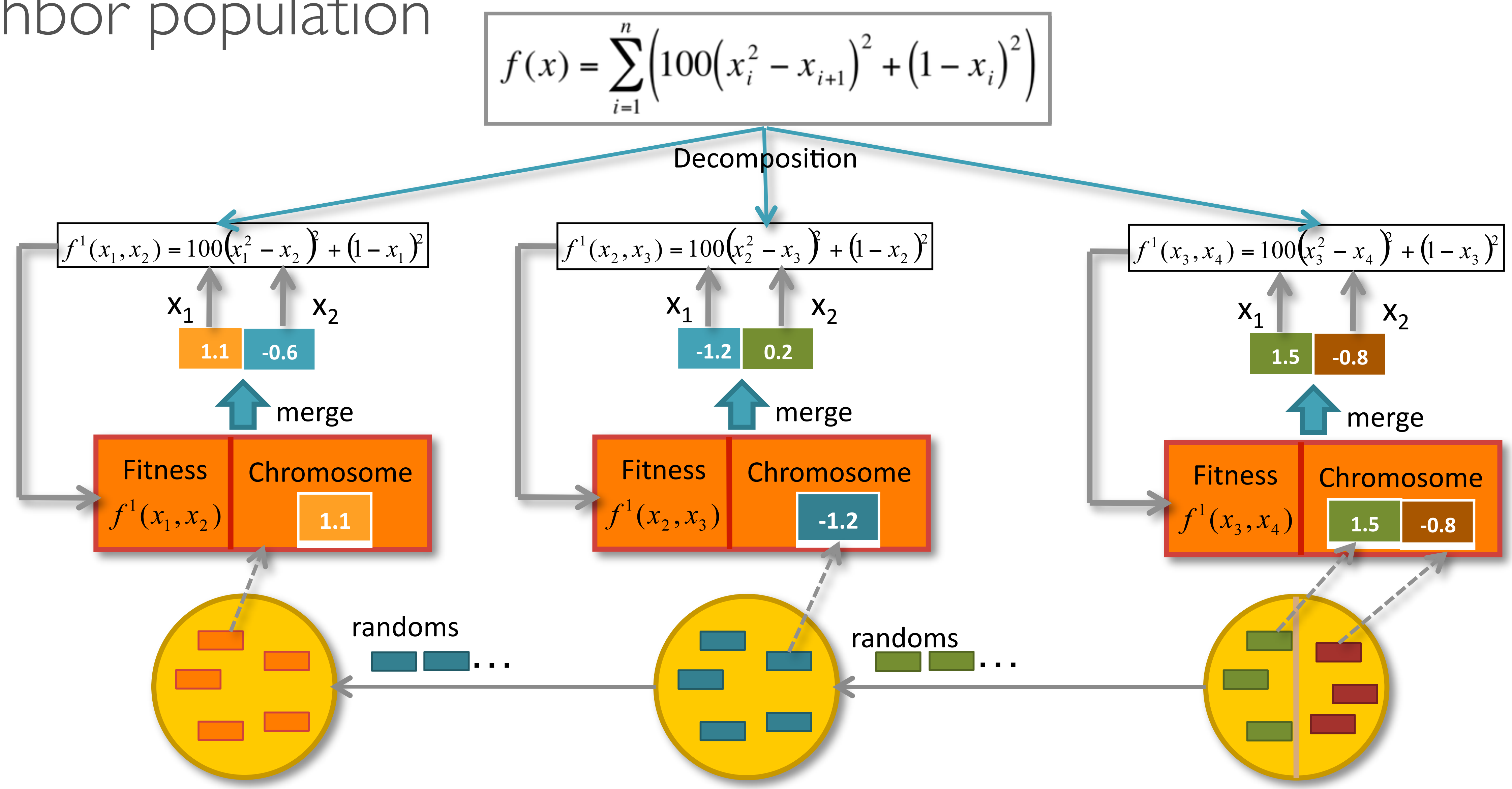
- A chromosome encodes a complete solution
- Solution evaluated on the global problem



$$f(x) = \sum_{i=1}^{n} \left(100\left(x_i^2 - x_{i+1}\right)^2 + \left(1 - x_i\right)^2 \right)$$

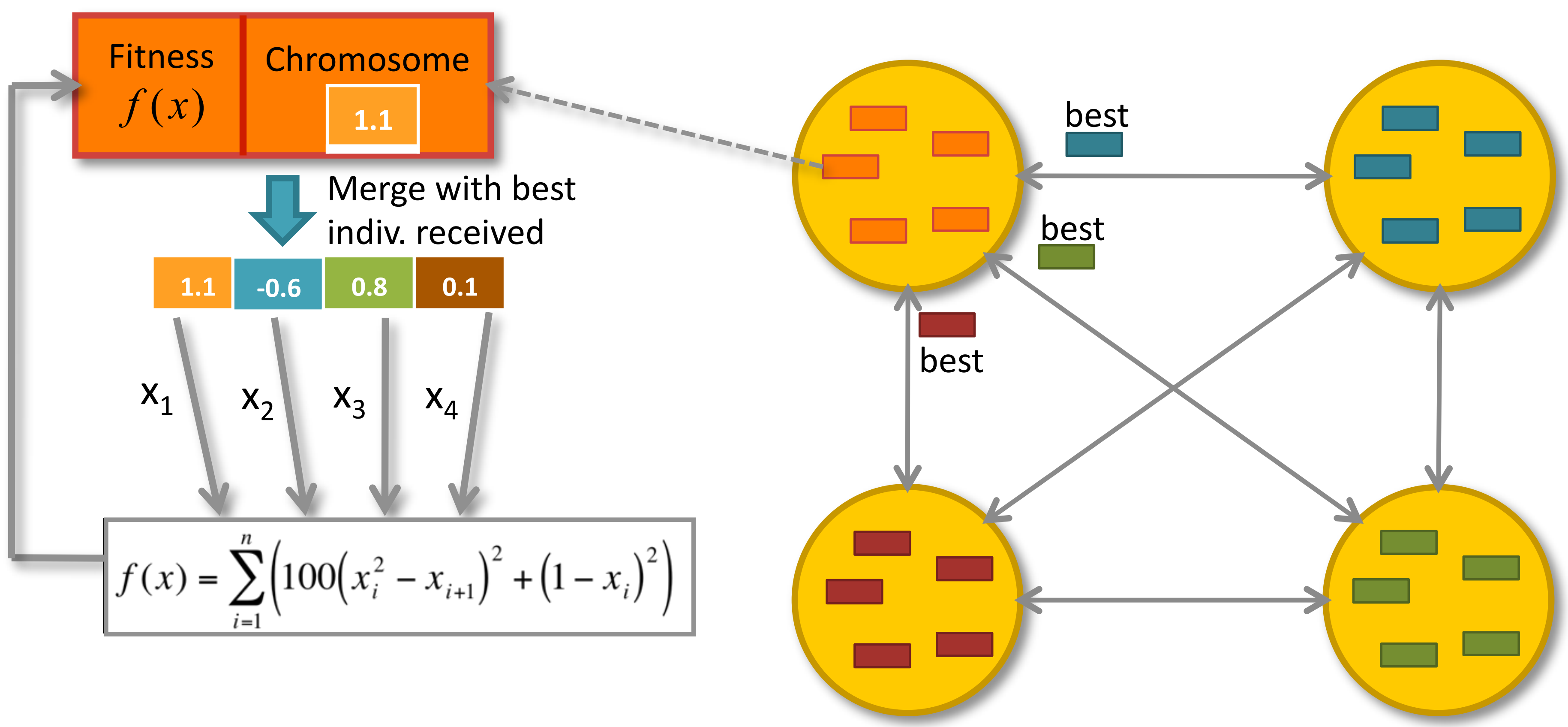
Loosely Coupled GA (LCGA)

- Each node runs a subpopulation for a **subset of the N variables**
- Each population **evaluates** its individuals on a **local subproblem** using a random individual received from its neighbor population



Cooperative Coevolutionary GA (CCGA)

- Each node runs a subpopulation for a **subset of the N variables**
- Each population **evaluates** each of its individuals **on the global fitness function** using the best individual received from each other subpopulation



Multi-objective Optimization

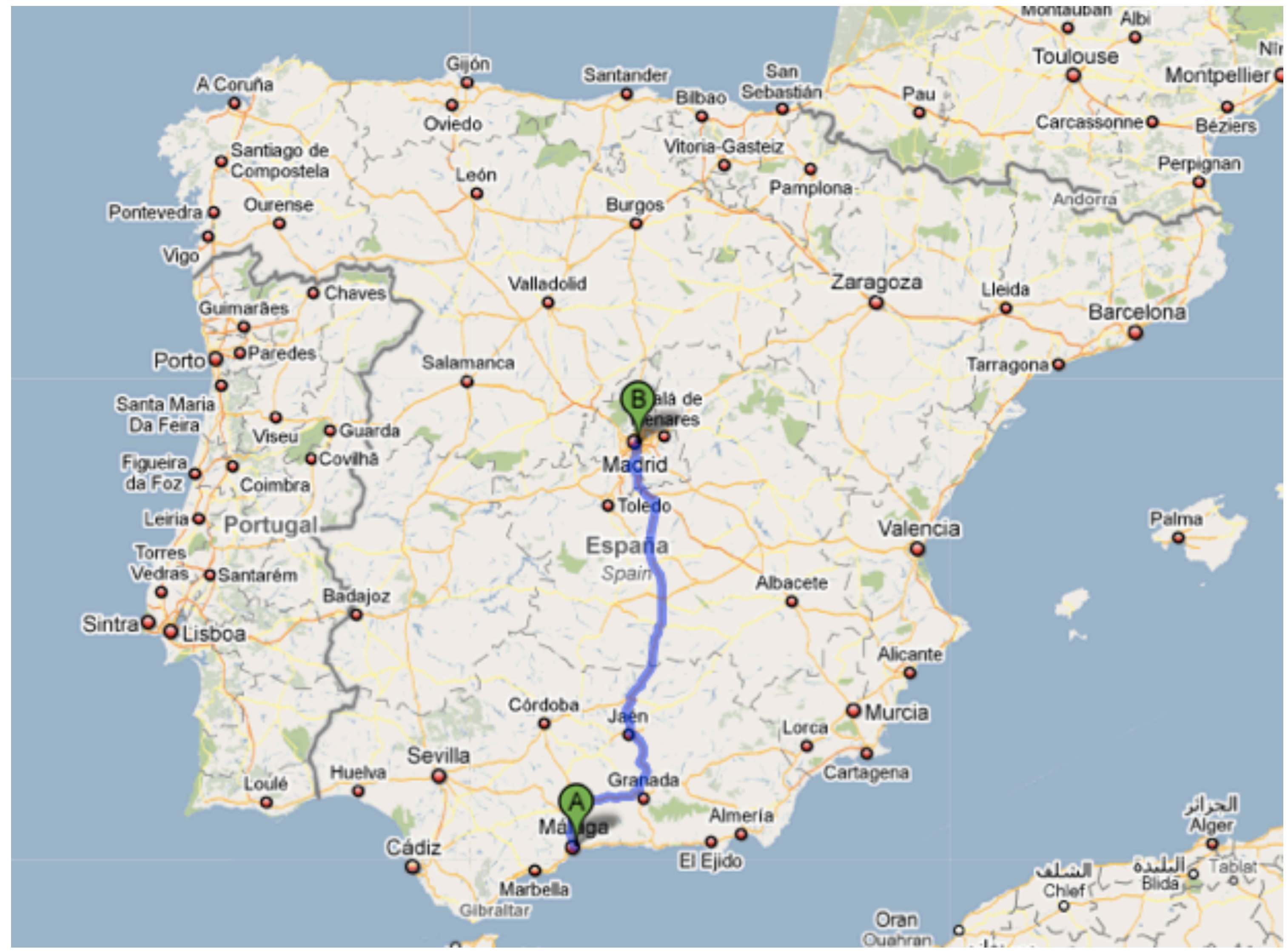
Multi-objective optimization

- Many real-world optimization problems require to optimize more than one objective at the same time
  - These objectives are usually **in conflict** among them
  - Improving one means worsening the others
- **Multi-objective** (or multi-criteria) optimization
  - Discipline focused on solving multiobjective optimization problems (MOPs)

$$\begin{array}{lll} \textit{Optimize} & f_m(\vec{x}) & m = 1, 2, ..., m \\ \textit{Subject to} & g_j(\vec{x}) \geq 0 & j = 1, 2, ..., j \\ & h_k(\vec{x}) = 0 & k = 1, 2, ..., k \end{array}$$

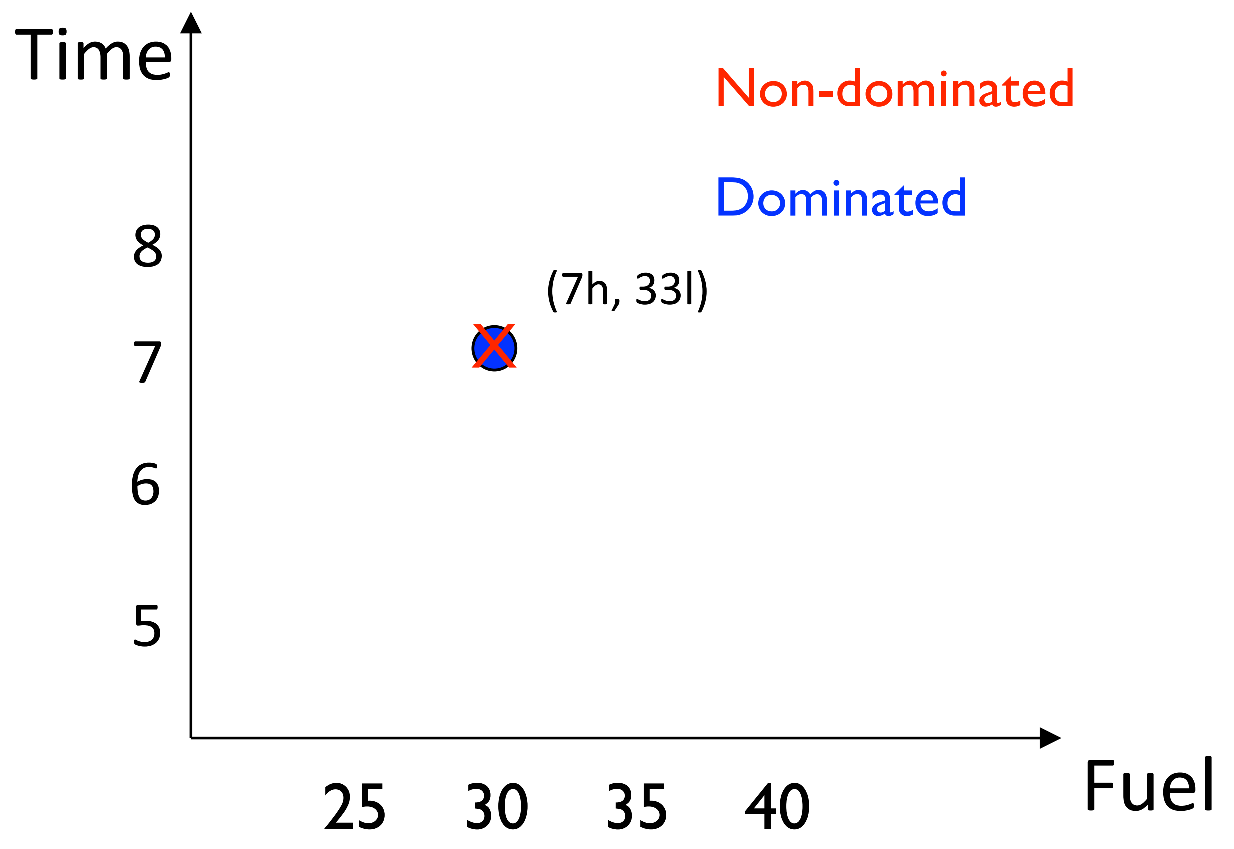
MO optimization: example

- Example: travelling by car from Málaga to Madrid (535 km)
  - Objective 1:
    - Minimizing time
  - Objective 2:
    - Minimizing fuel
  - Constraints:
    - Max. speed: 120 km/h
    - Min. speed: 60 km/h
  - Decision variable:
    - mean car speed



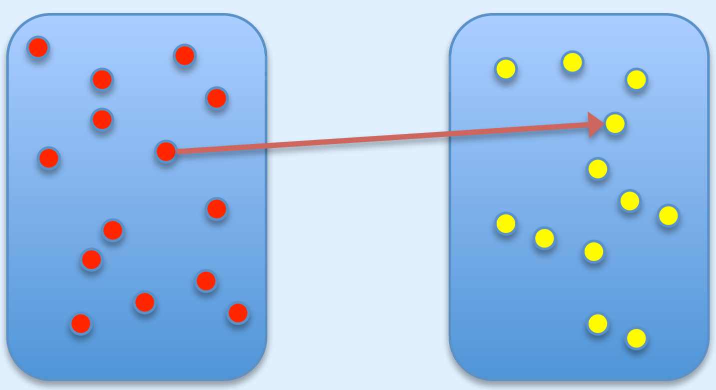
MO optimization: example

- Travelling by car from Málaga to Madrid (535 km)
  - Extreme solutions
    - Time: 5 hours, fuel: 9.0 litres
    - Time: 8 hours, fuel: 6.0 litres
  - Other solutions
    - Time: 5.5 hours, fuel: 7.5 litres
    - Time: 6 hours, fuel: 6.5 litres



Multi-objective optimization

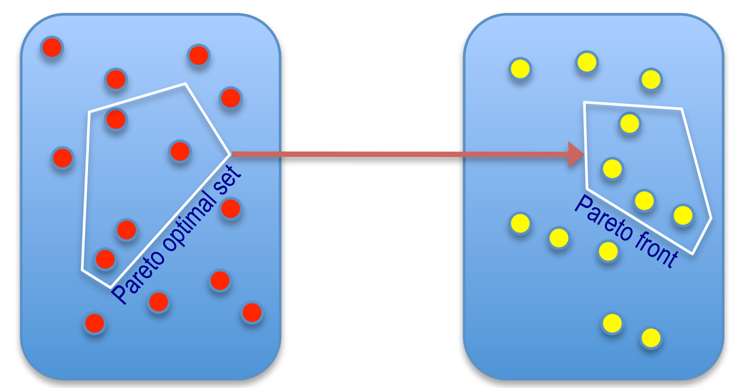
- In single-objective optimization (SO)
  - The optimum is
    - One solution
    - Several ones with same quality



X
(Solution space)

F(X)
(Objective space)

- In multi-objective optimization (MO)
  - The optimum (Pareto optimal set) is a set of (non-dominated) solutions



X
(Solution space)

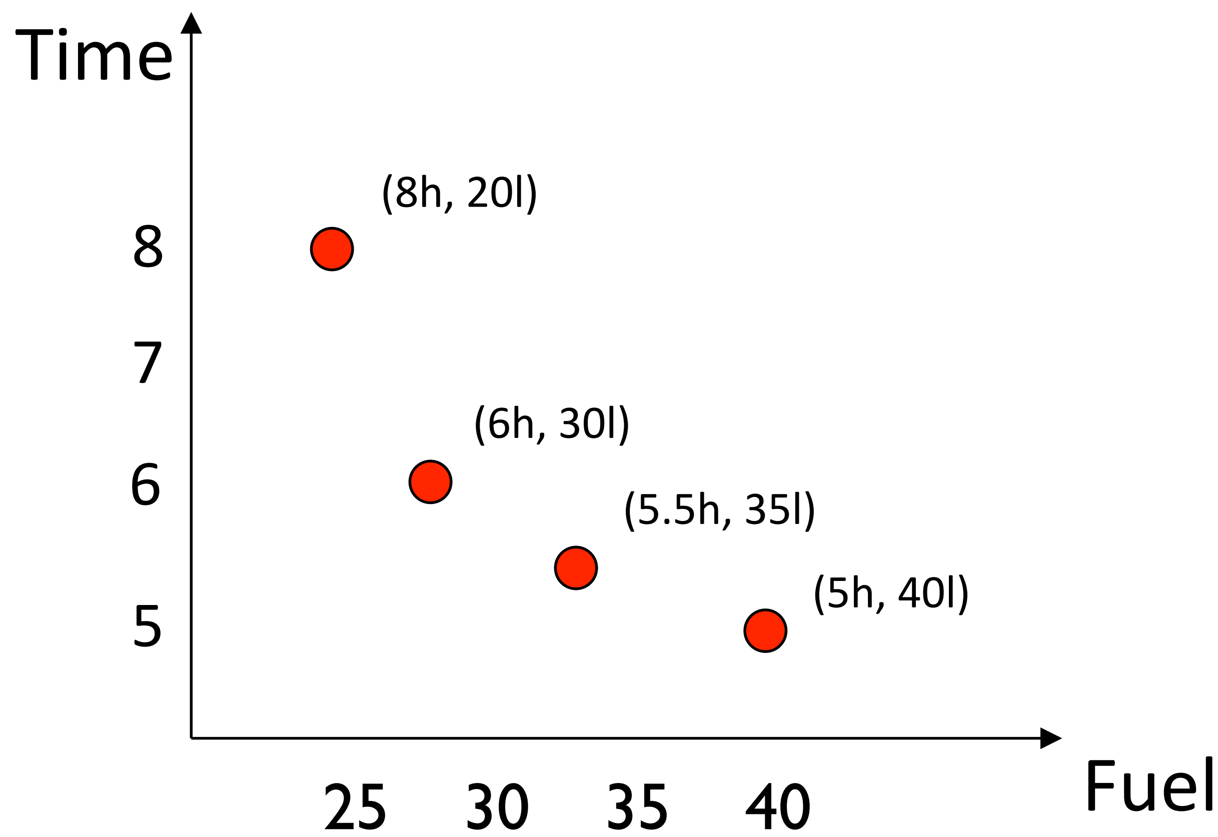
F(X), G(X), …
Objective space

MO Optimization and Decision Making

- Finding the Pareto front of a problem is not the last step in multi-objective optimization

- In practice, an expert in the domain (the decision maker) has to choose the best trade-off solution

- In the example of traveling from Málaga to Madrid
- If time is important
  - Choose (5h, 40l)
- If consumption is important:
  - Choose (8h, 20l)
- Compromise solution:
  - (6h, 30l)
  - (5.5h, 35l)



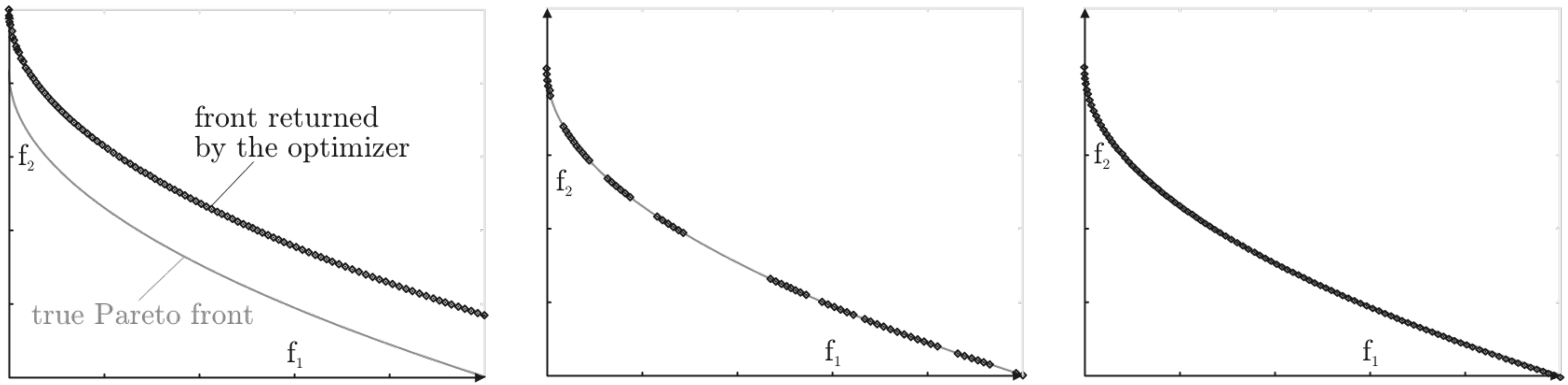
The goal of MO optimization

Universidad de Cádiz

- The ideal goal is to obtain the Pareto front
- Unfortunately, this is unpractical in real-world problems
  - NP-hard complexity, non-linearity, epistasis, ...
  - Frequently, exact techniques are not useful
- Alternative: Use non-exact algorithms
  - E.g. Metaheuristics
  - These techniques provide an approximation to the Pareto front

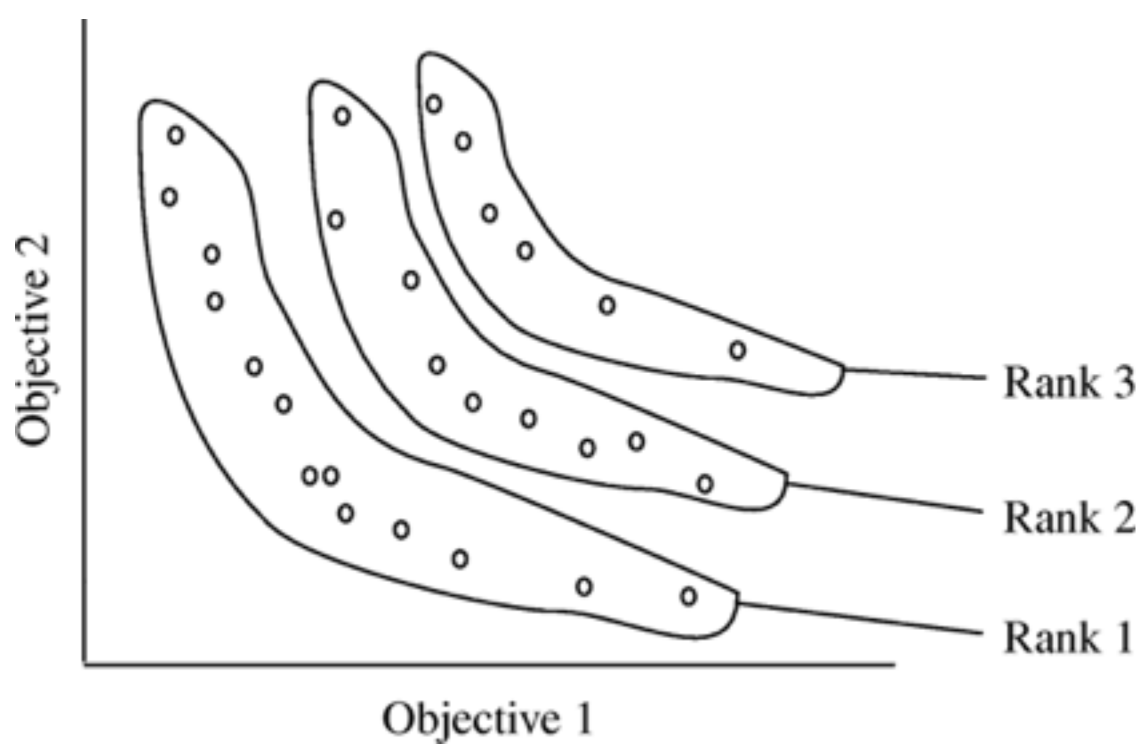
The goal of MO optimization

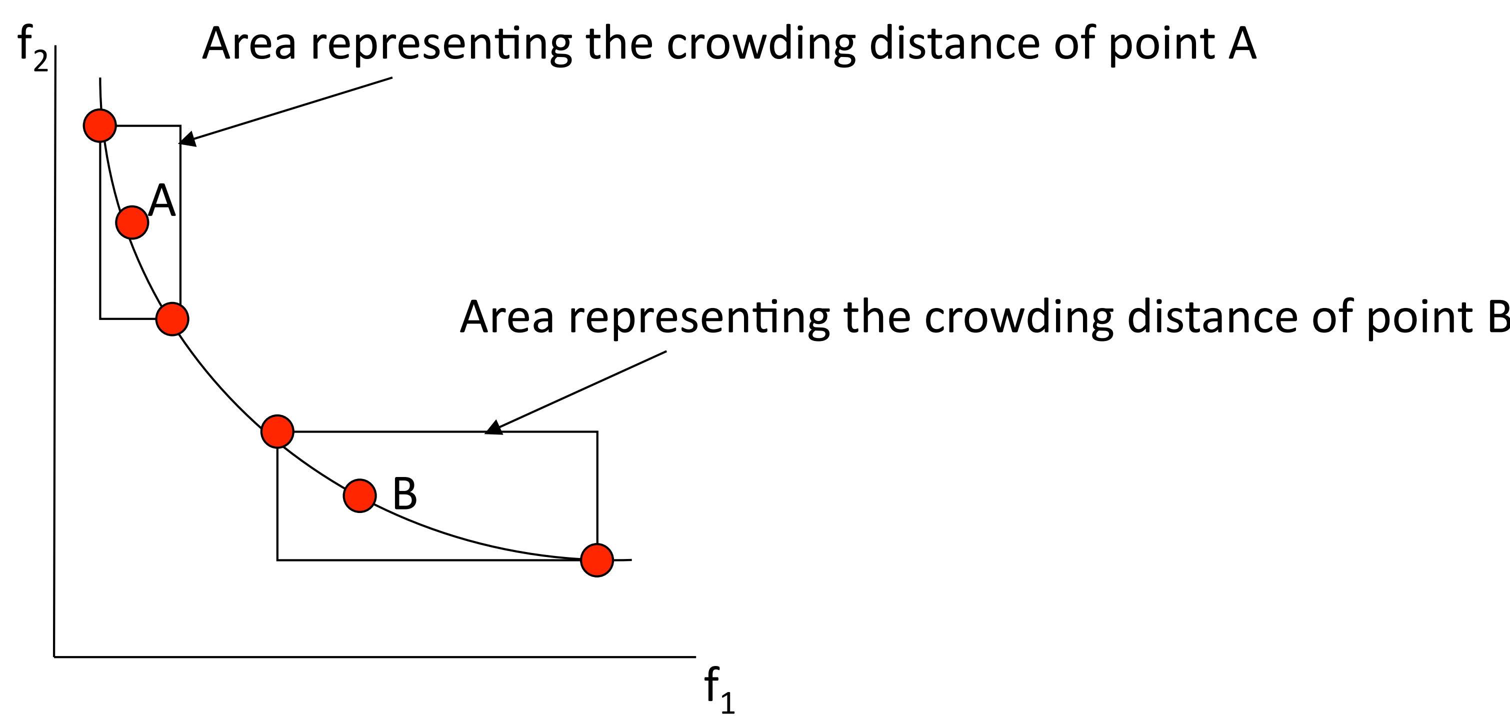
- The goal is to find the **Pareto front**
- Exact techniques are **not useful** in most cases
  - NP-hard complexity, non-linearity, epistasis , …
- Rely on **approximation** techniques
  - Two key features to measure the **quality** of solutions
    - Convergence
    - Diversity



NSGAII

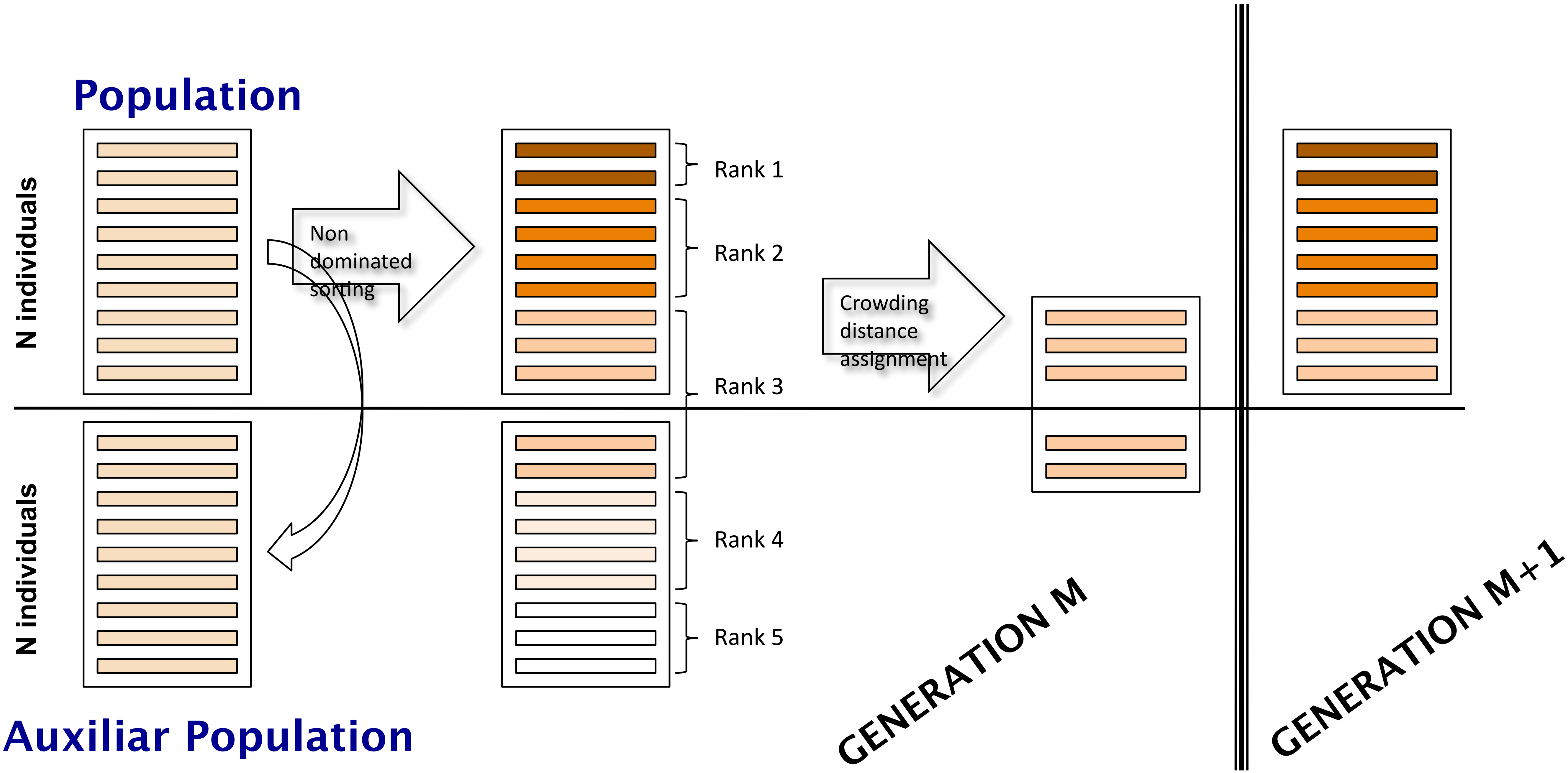
- Non-dominated Sorting Genetic Algorithm
- The most popular metaheuristic for multi-objective optimization
- Features
  - Ranking using non-dominated sorting
  - Crowding distance as density estimator





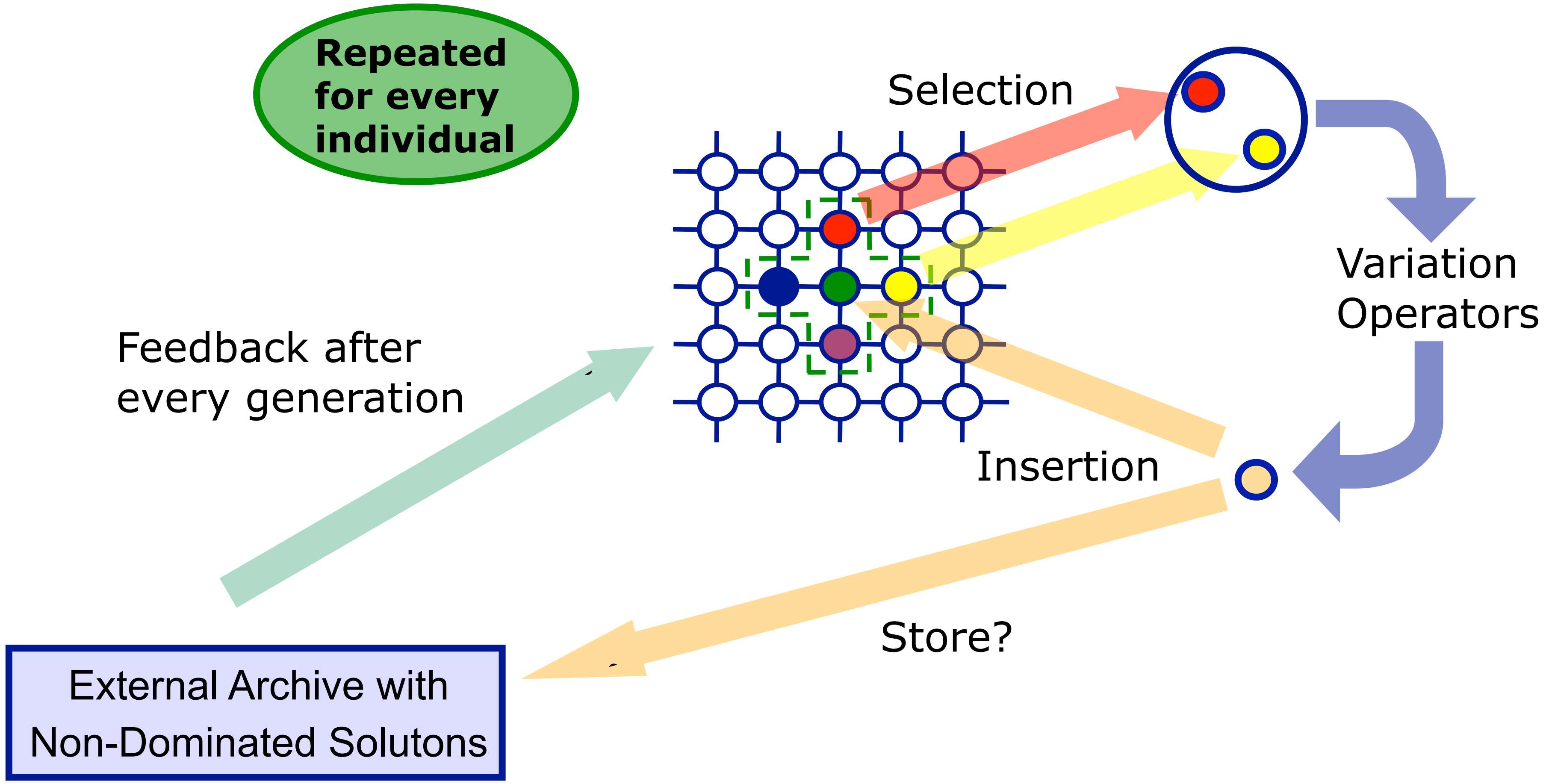
Point B is in a less crowded region than point A

NSGAII



Population
N individuals
Non dominated sorting
Rank 1
Rank 2
Rank 3
Rank 4
Rank 5
Crowding distance assignment
N individuals
Auxiliar Population
GENERATION M
GENERATION M+1

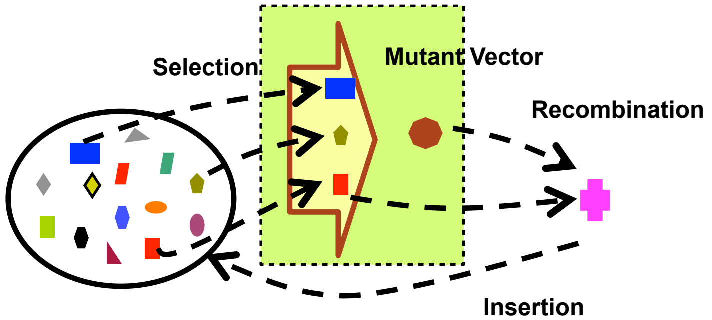
CellDE

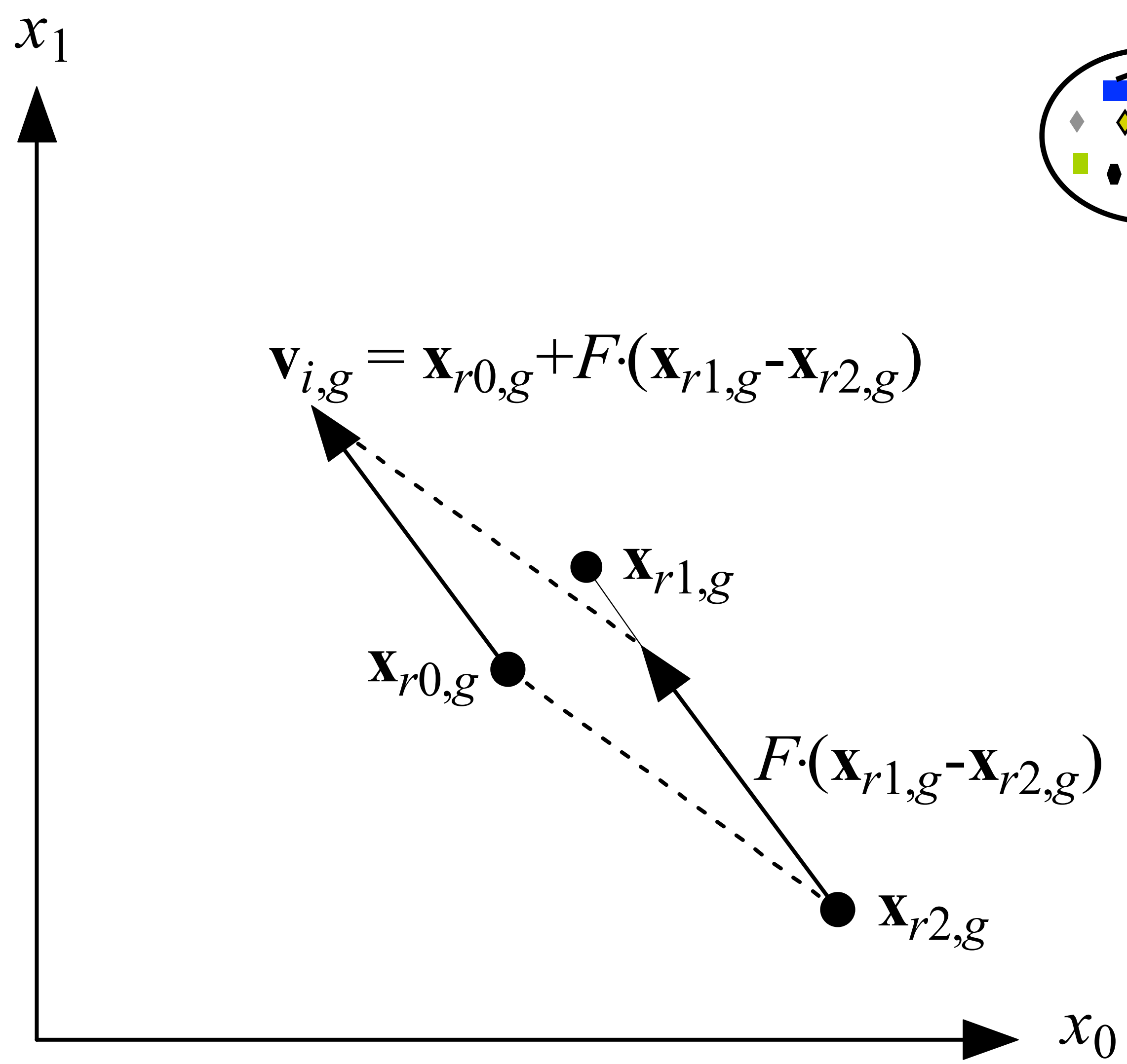


Repeated for every individual
Selection
Variation Operators
Feedback after every generation
Insertion
Store?
External Archive with Non-Dominated Solutons

Differential evolution

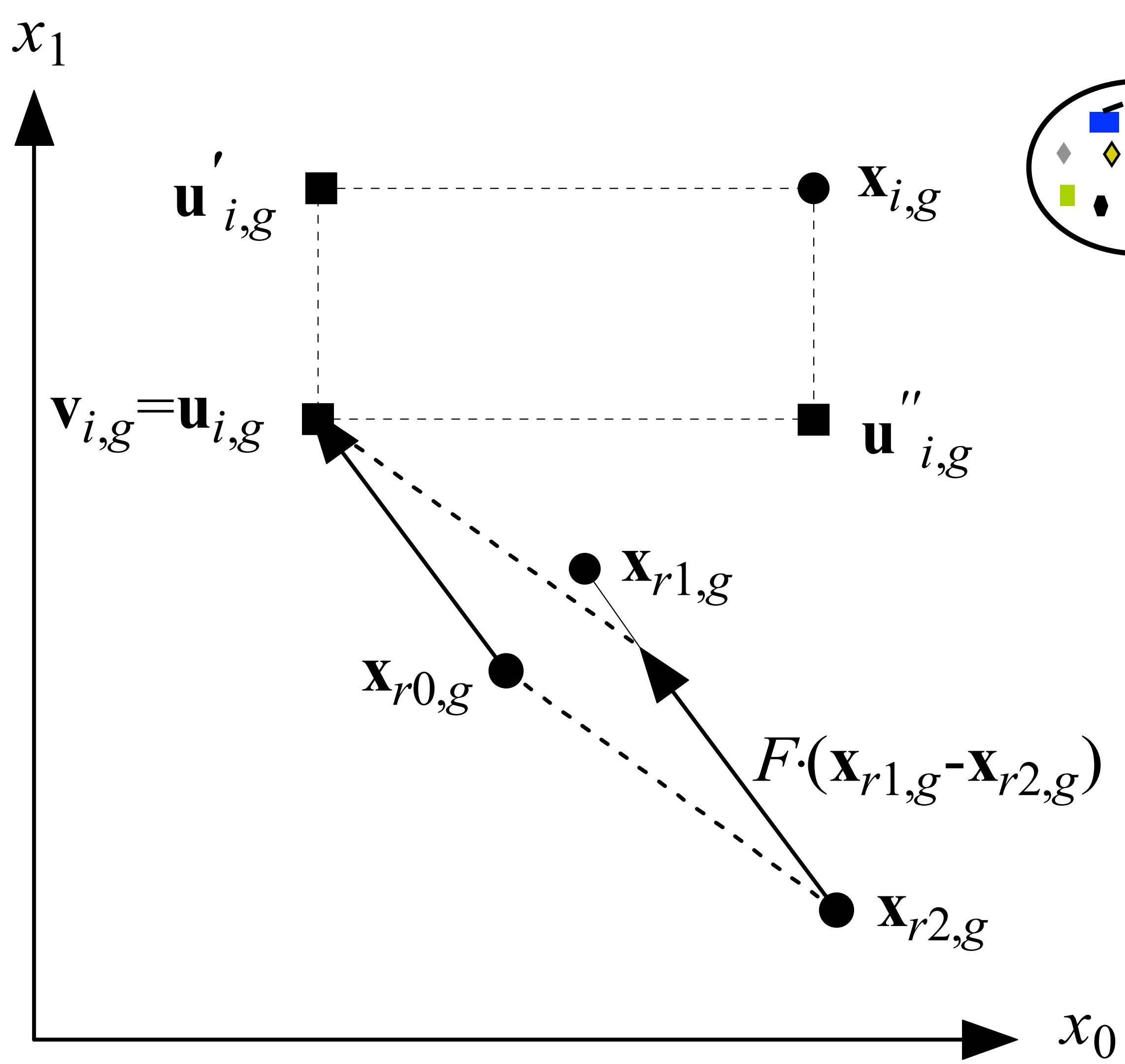
- Mutant vector generation

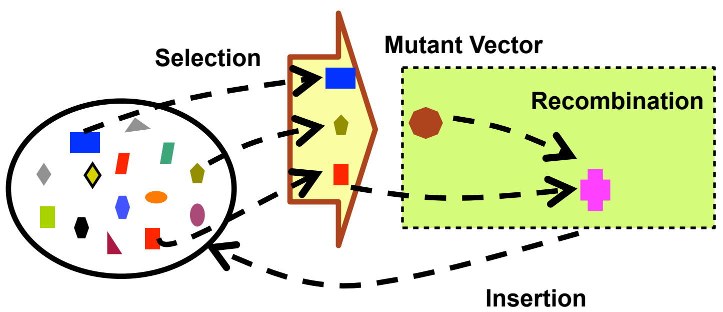




Differential evolution

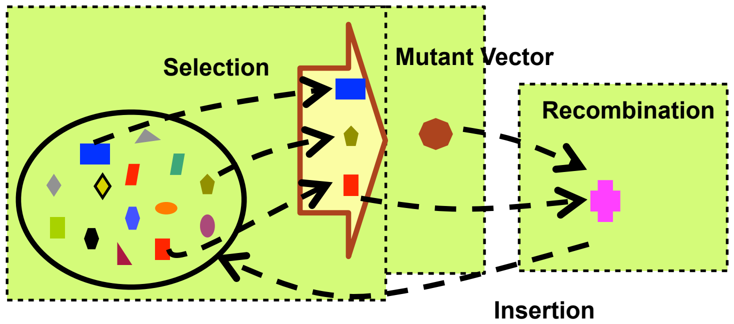
- Recombination





Differential evolution

$x_{r0,g}$	1	5	-2,5	30,7	0,4
$x_{r1,g}$	5,7	8,9	0,3	-4,3	30,5
$x_{r2,g}$	4,2	7,2	3,3	8,2	6,7



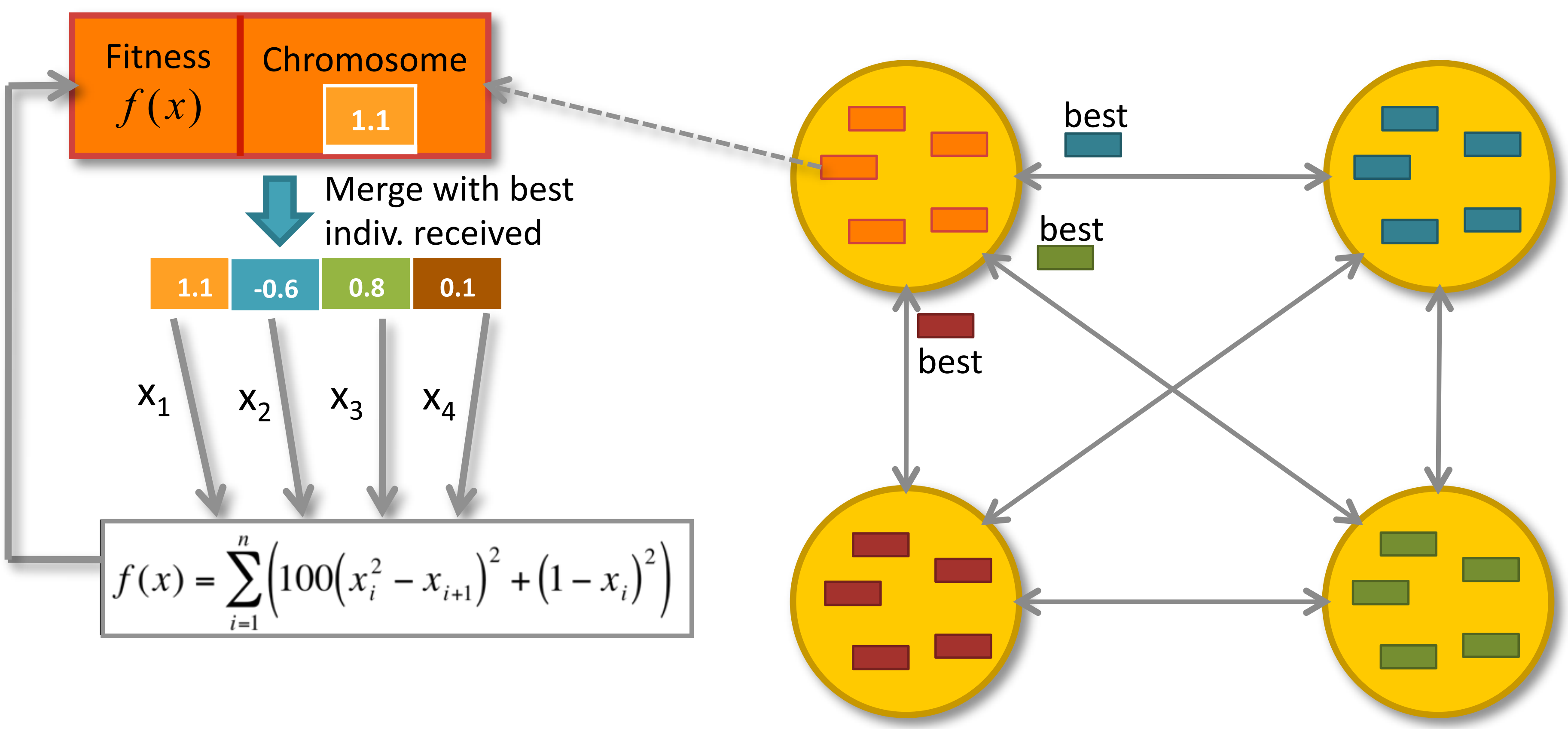
$v_{i,g}$	1,75	5,85	-0,7	24,45	12,3
$x_{r0,g}$	1	5	-2,5	30,7	0,4

$$u_{i,g} = \begin{cases} v_{i,g} & \text{With prob. CR} \\ x_{r0,g} & \text{otherwise} \end{cases}$$

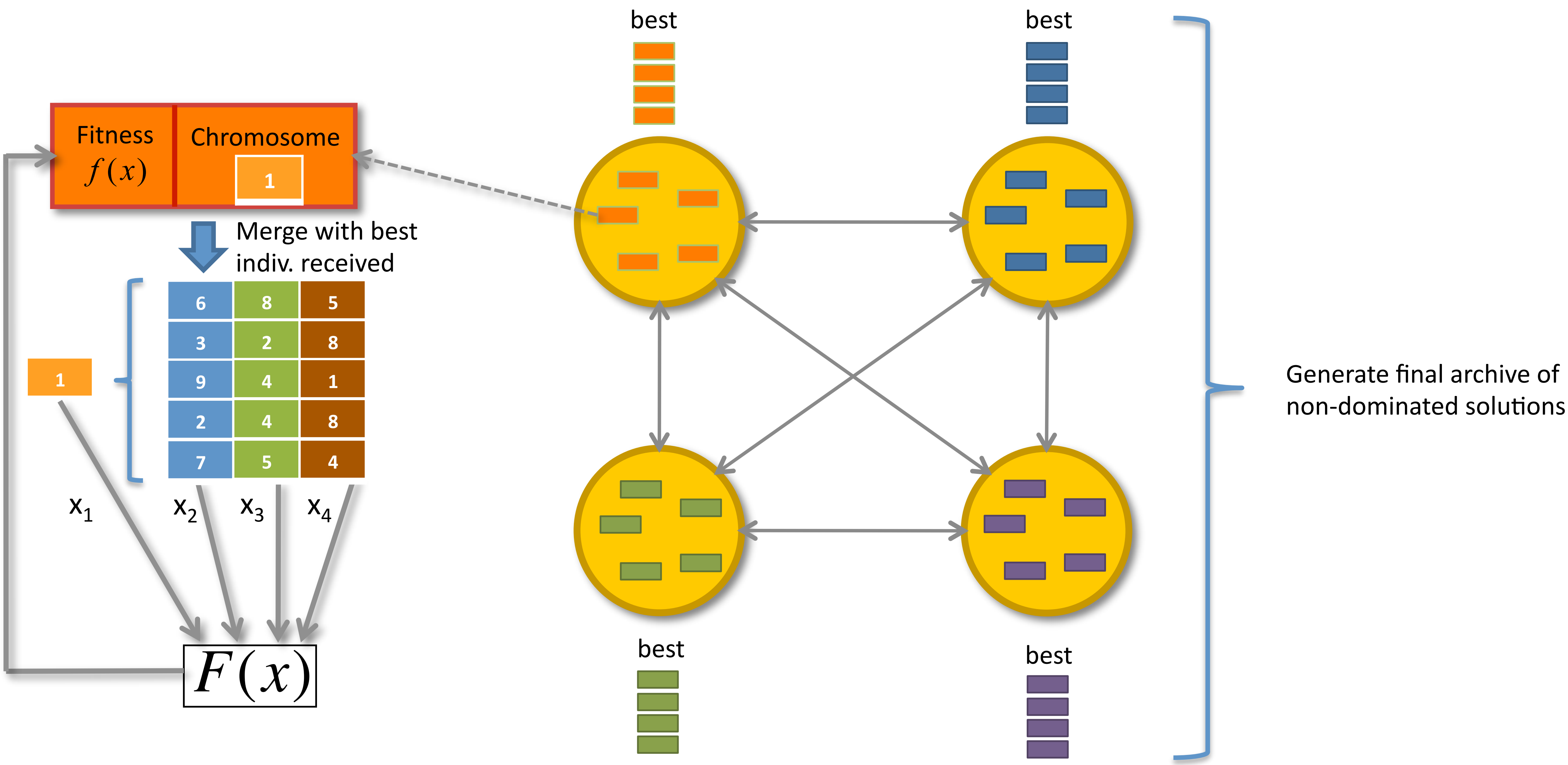
$u_{i,g}$	1,75	5	-0,7	30,7	0,4

Cooperative Coevolutionary GA (CCGA)

- Each node runs a subpopulation for a **subset of the N variables**
- Each population **evaluates** each of its individuals **on the global fitness function** using the best individual received from each other subpopulation



Multi-objective CCGA



Three new multi-objective CCGAs

- Three CCMOEAs designed
  - Based on NSGA-II: CCNSGAII
  - Based on SPEA2: CCSPEA2
  - Based on MOCell: CCMOCell

NSGA-II

- Reference algorithm
- Panmictic population
- Selection of solutions
  - Ranking
  - Crowding

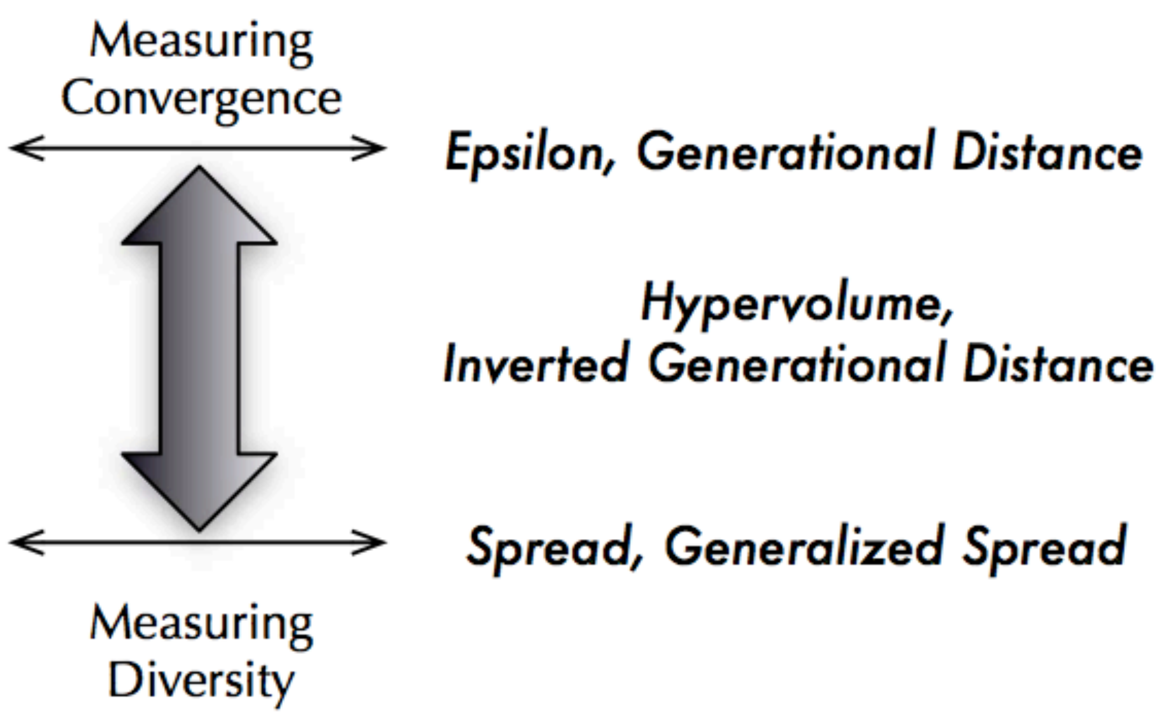
SPEA2

- Panmictic population
- External archive
  - Strength raw fitness
  - k-nearest neighbors

MOCell

- Cellular population
  - Only next individuals can interact
- External archive
  - Feedback to population

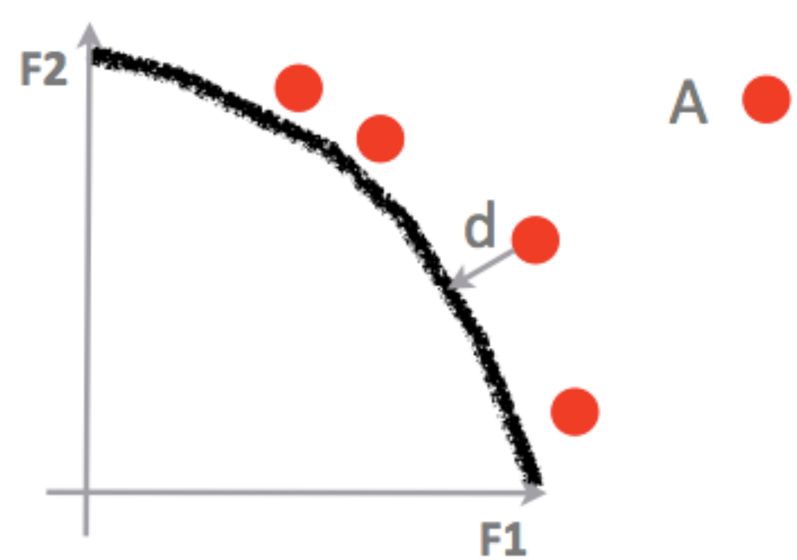
Quality metrics



Quality metrics

- Generational distance
  - Average distance of every solution of a front A to the Pareto front
  - Convergence to the true Pareto front

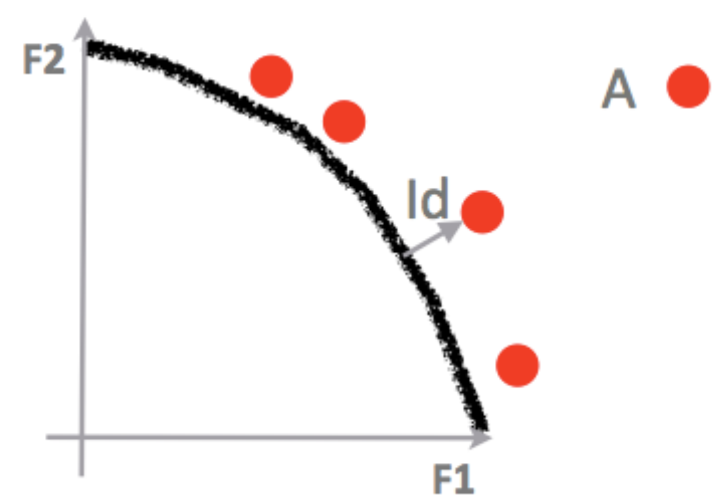
$$GD(A) = \frac{\sum_{1..n} d}{n}$$



Quality metrics

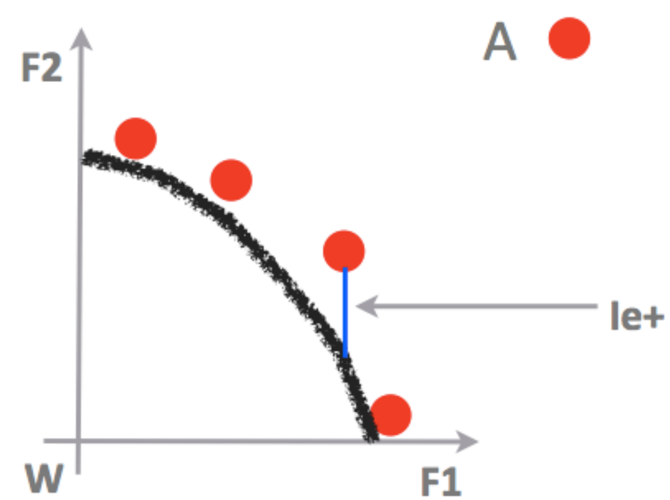
- Inverted generational distance
  - Average distance of every solution of the points of the Pareto front to of a front A
  - Convergence to the true Pareto front

$$IGD(A) = \frac{\sum_{1..n} Id}{n}$$



Quality metrics

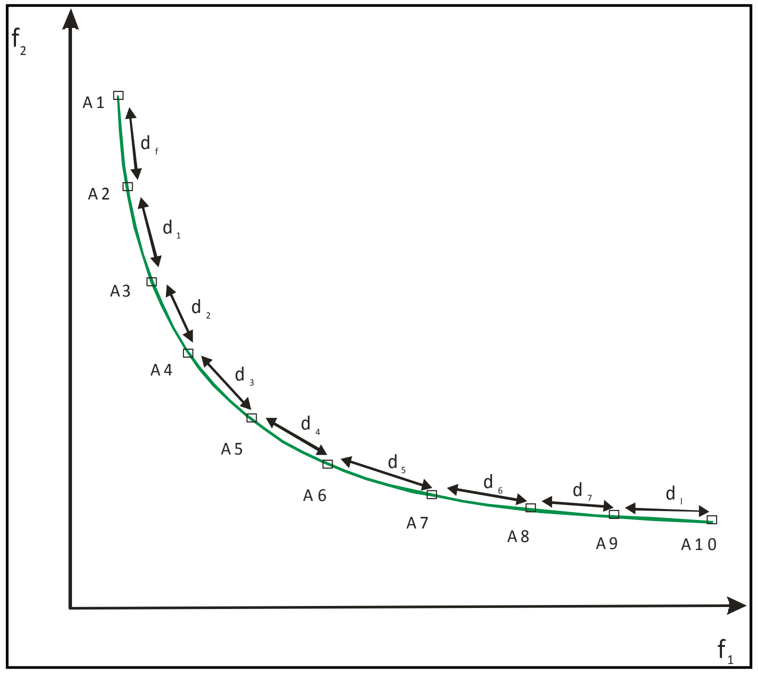
- Additive epsilon indicator
  - Convergence to the Pareto front
  - Given an approximation set A, this indicator is a measure of the smallest distance we would need to translate every solution in A so that it dominates the Pareto front



Quality metrics

- Spread
  - Diversity of the solutions along the Pareto front

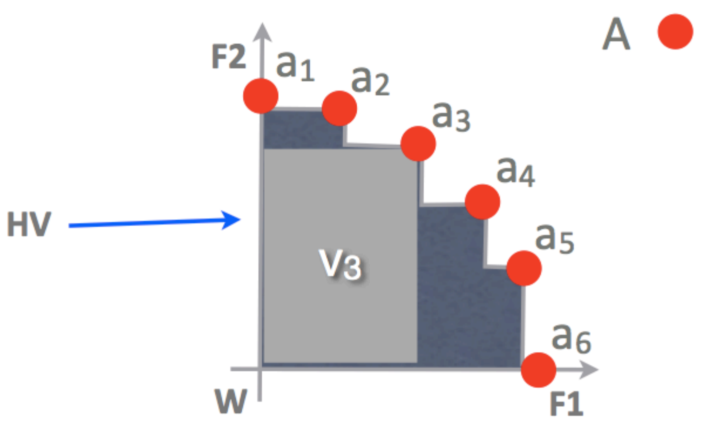
$$\Delta = \frac{d_f + d_l + \sum_{i=1}^{N-1} \left| d_i - \bar{d} \right|}{d_f + d_l + (N-1)\bar{d}}$$



Quality metrics

- Hypervolume
  - Takes into account both convergence and diversity
  - Measures the region dominated by the obtained font

$$HV = volume\left(\bigcup_{i=1}^{|Q|} v_i\right)$$



Quality metrics

- All discussed metrics require the optimal Pareto front
  - Either for computations
  - Or to normalize the fronts
- What if we do not know it?
  - Build a reference Pareto front of (hopefully) quasi-optimal solutions
    - Run the problem with different algorithms
    - Run every algorithm a large number of times
    - Take the best non-dominated solutions found in all runs by all algorithms

Methodology for comparing algorithms

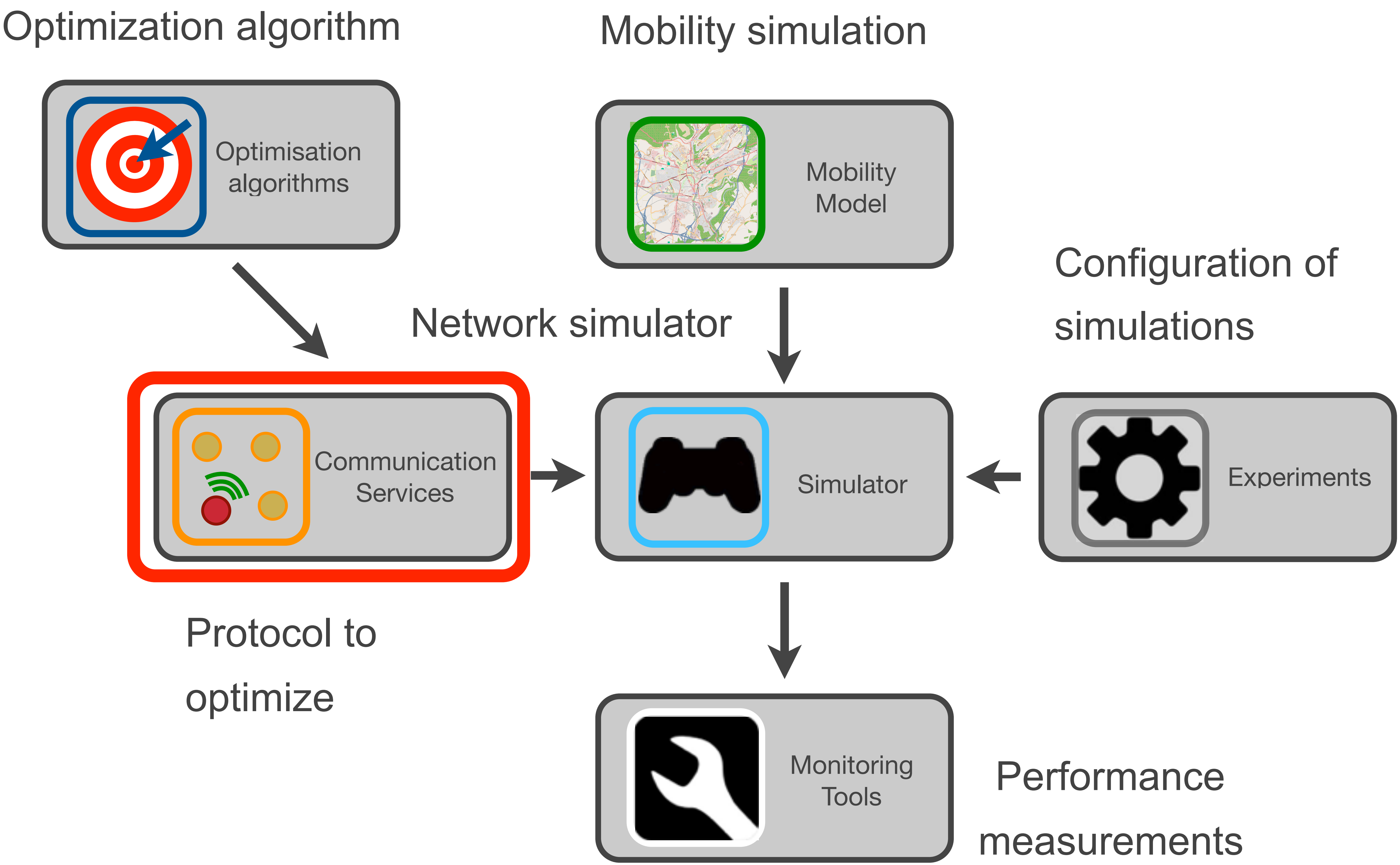
- Metaheuristics are stochastic algorithms
  - Repeating the same experiment may lead to different results
  - It is necessary to apply a rigorous statistical methodology to assess the performance of a metaheuristic
- To draw firm conclusions, we need to look for statistical significance on the results

Methodology for comparing algorithms

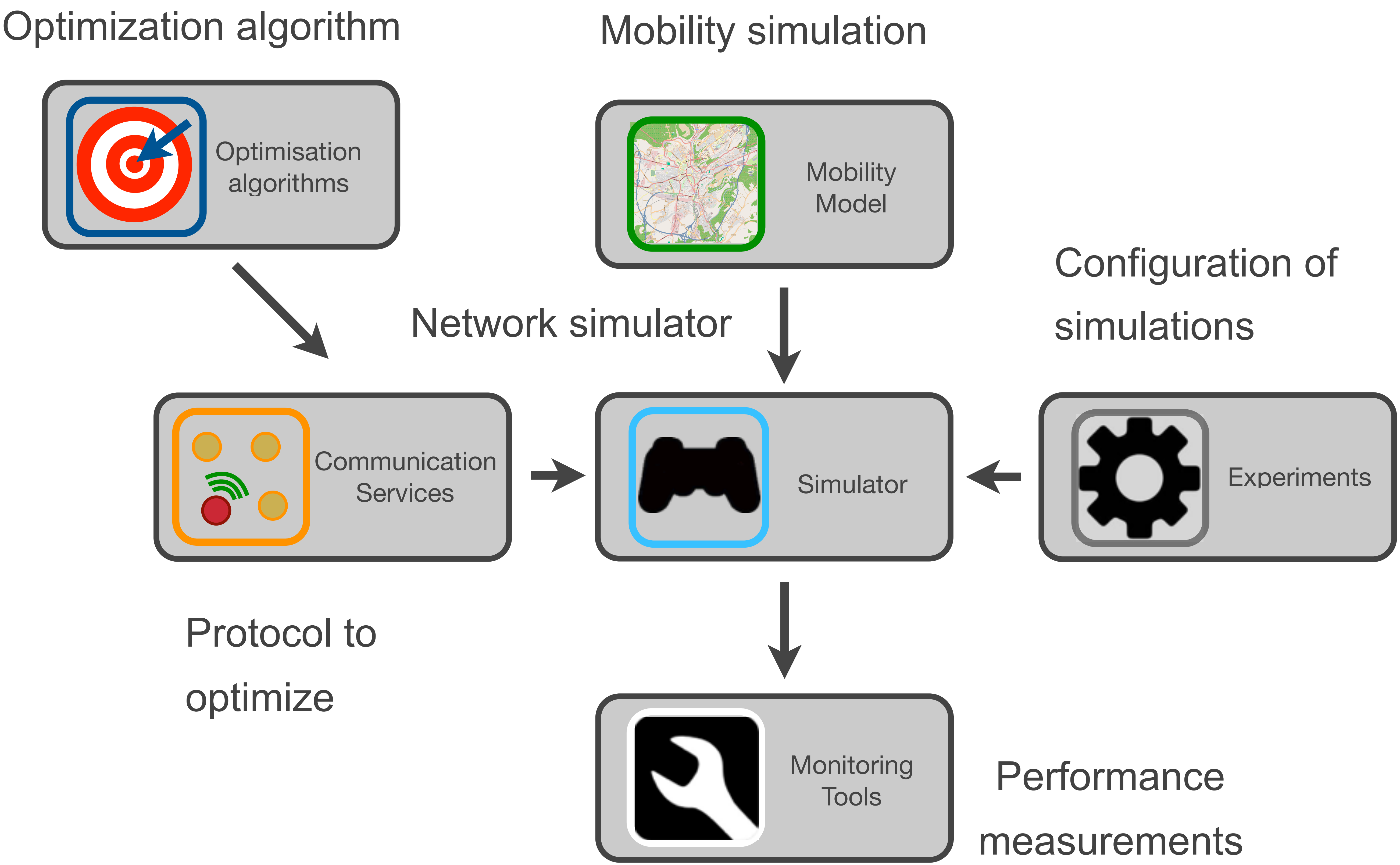
- Statistical significance
  - Large number of independent runs
  - Compute quality metrics
  - Statistical test on the results of the quality metrics
    - Non-parametric test: Wilcoxon unpaired signed-ranks test
  - Confidence level of 95%
    - Significance level of 5% or p-value under 0.05 in the statistical tests
    - This means that the differences are unlikely to have occurred by chance with a probability of 95%

Optimization Framework for Mobile Ad Hoc Networks

AEDB: Experimental Framework



Off-line optimization process



On-line optimization process