GENETIC ALGORITHM

ANALYSIS AND EFFECTIVENESS EVALUATION IN SOLVING REAL-WORLD PROBLEMS

Department of Electronics and Telecommunications
01RGBRV Optimization methods for engineering problems
Prof. M. Repetto

Antonio Calagna, Giuseppe Di Giacomo, Francesco Gabriele and Deborah Volpe

June 7, 2022

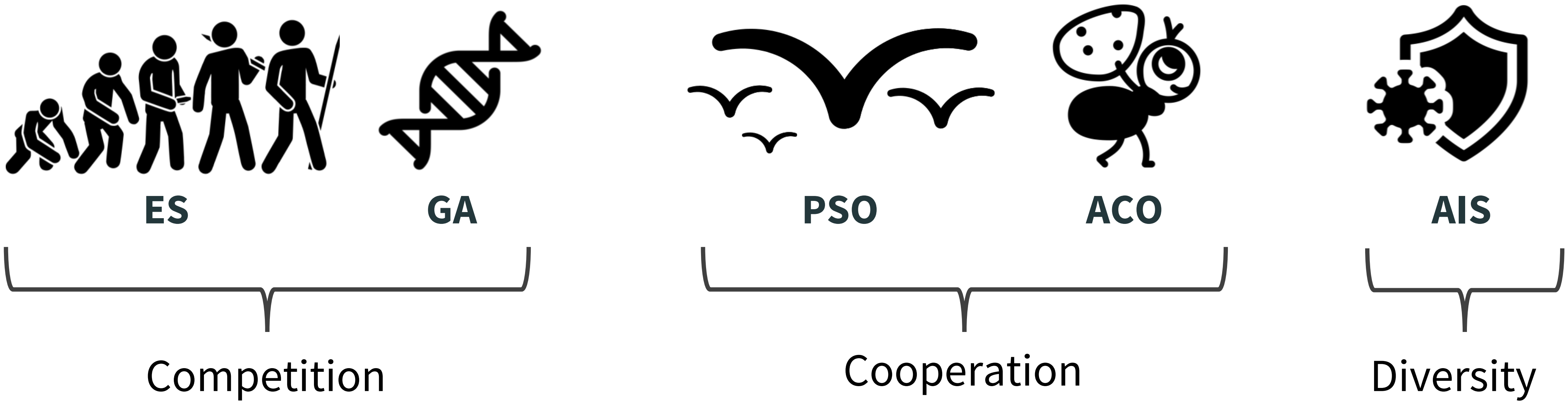
Outline

1. Introduction
2. Optimal Placement of antennas in an array
3. Application to QUBO problems
4. Conclusions

PART I: Introduction

Stochastic Optimization Algorithms

- Stochastic optimization has become the reference approach for engineering problems
- Most algorithms have been developed by implementing analogies with natural phenomena



[1] Icons provided by: Jackob Vogel, Debruder studio, Waleed Egalamy, Orin Zuu, Adrien Coquet at thenounproject.com

EVOLUTIONARY ALGORITHMS

- Direct, parallel, stochastic methods for global search and optimization, which imitate the evolution of the living beings, i.e., reproduction, natural selection and diversity
- Based on the competition among individuals in the population

Population: a group of individuals, each one with its properties
Fitness: measure of the level of adaptation
Genetic outfit: heritage content
Genetic operators: outfit update by means of crossover and mutation
Selection: which individuals move to the next generation

ES

GA

Evolutionary Algorithms

ES

Floating Point
Deterministic
Changing
Sequential

GA

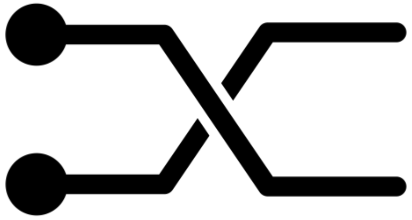
Binary
Probabilistic
Fixed
Parallel

Representation
Selection
Genetic Operators
Algorithm Execution

Genetic Algorithm

- **Representation**: traditionally, individuals are defined as binary strings
  - Chromosomes: strings, candidate solutions
  - Genes: alphabets
  - Alleles: values of genes

- **Mating pool**: individuals that will take part to reproduction
  - Roulette Wheel: probabilistic selection



Cross-over: two parents mingle their genetic outfit

Exploitation

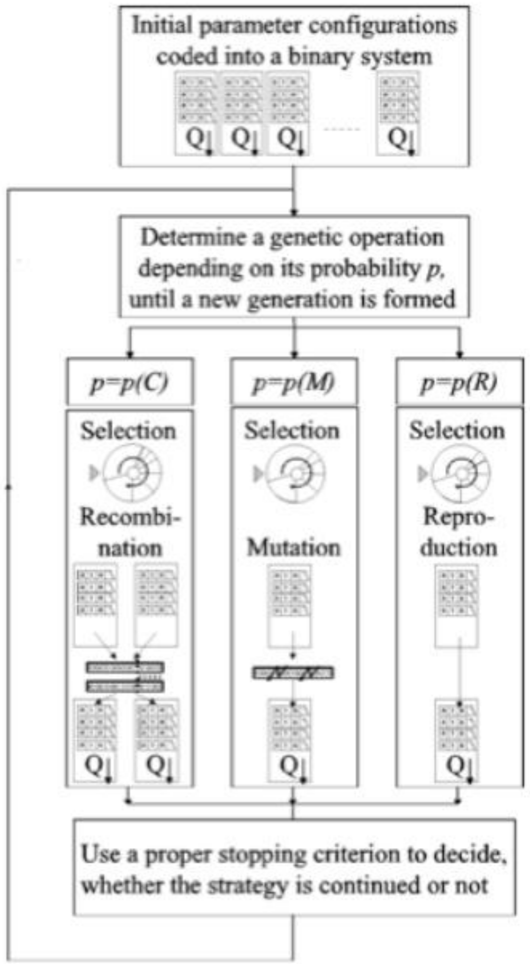
Mutation: random change to alleles

Exploration

[2] Icons provided by Federico Panzano, Med Marki, Kamin Ginkaew, Joe Harrison, Debruder studio at thenounproject.com

Genetic Algorithm

- **Initialization:** the initial population is generated randomly
- **Evaluation**: the fitness values of the candidate solutions are evaluated
- **Selection**: survival-to-the-fittest mechanism is applied on the candidates
- **Recombination**: combine two or more parental solutions to generate offspring
- **Mutation**: local and random modification of a candidate
- **Replacement**: the offspring population replaces the original parental one



GENETIC ALGORITHM

Sastry, K., Goldberg, D., & Kendall, G. (2005). Genetic algorithms. In *Search methodologies* (pp. 97-125). Springer, Boston, MA.

- **Genetic Operators:**
  - **Selection Operators:**
    - Fitness Proportionate Selection
    - Ordinal Selection
  - **Crossover Operators:**
    - K-point Crossover
    - Uniform Crossover
    - Uniform Order-based Crossover
    - Order-based Crossover
    - Partially Matched Crossover
    - Cycle Crossover
  - **Mutation Operators:**
    - Bit-flip Mutation
    - Problem-specific strategies

- **Replacement:**
  - Delete-all
  - Steady-state
  - Steady-state-no-duplicates

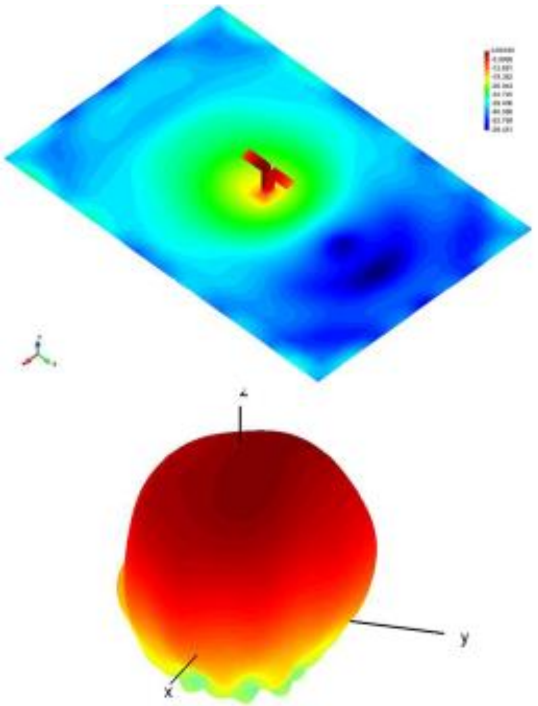
- **Efficiency/Effectiveness Optimization:**
  - Parallelization
  - Hybridization
  - Time continuation
  - Evaluation relaxation

- **Competent Genetic Algorithms:**

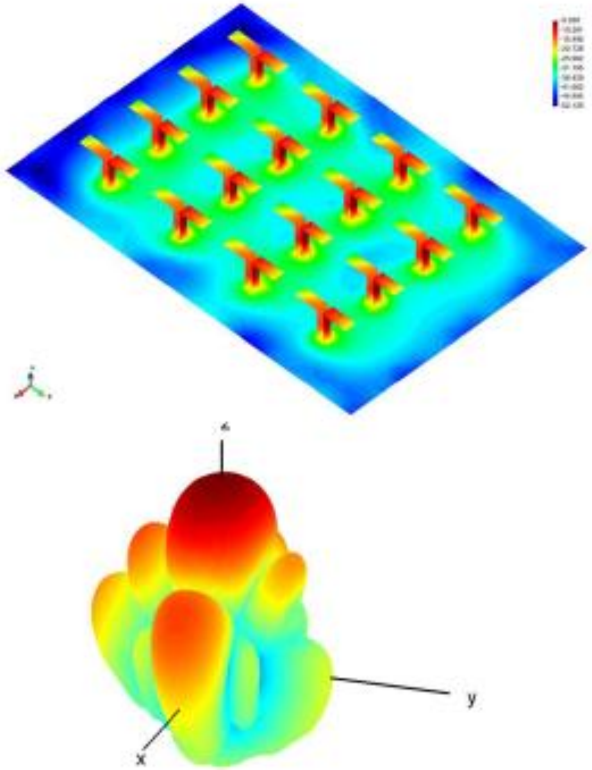
Solve hard problems, quickly, reliably and accurately by automatically adapting problem, coding and operators

PART II: Optimal placement of antennas in an array

Optimal placement of antennas in an array - I



Antenna Element



Antenna Array

Optimal placement of antennas in an array - II

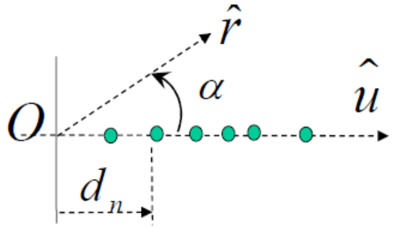
To study the array radiation phenomena, some approximations need to be done:

- Inter-element coupling in radiation is neglected.
- The observation point P is seen under a small (solid) angle. All observation directions are approximatively equal (*Far-field approximation*).
- Identical, equi-oriented and equipolarized radiating elements.

$$\underline{E}(r,\theta,\varphi) \approx \underline{E}_0(r,\theta,\varphi)AF(\theta,\varphi)$$

$$AF(\theta,\varphi) \equiv \sum_n a_n \exp(jk\hat{r}\cdot\underline{r}_n) \qquad a_n = \frac{I_n}{I_1}$$

Focusing on a **Linear 1D Array**:

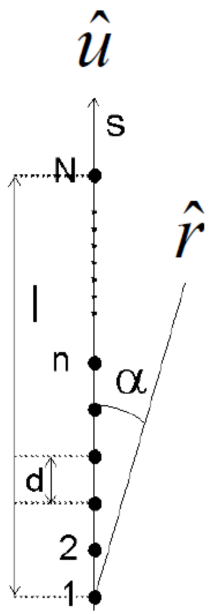


$$AF(\theta,\phi) = \sum_{n=1}^{N} I_n \exp(+jk\hat{r}\cdot\hat{u}d_n)$$

$$\hat{r}\cdot\hat{u} = \cos\alpha(\theta,\phi)$$

Optimal placement of antennas in an array - III

1D Equispaced Uniform Phased Arrays



$$AF(\theta,\phi)=\sum_{n=1}^{N} I_n \exp(+jk\hat{r}\cdot\hat{u}d_n)$$

$$\hat{r}\cdot\hat{u}=\cos\alpha(\theta,\phi)$$

- Uniform amplitude:

$$|I_n|=1\ \forall n, \quad I_n=\exp(j\Phi_n)$$

- Uniform phase difference between neighboring elements:

$$\angle I_n-\angle I_{n-1}=const.=\Phi$$

$$I_n=\exp(j(n\text{-}1)\Phi)$$

- The AF becomes:

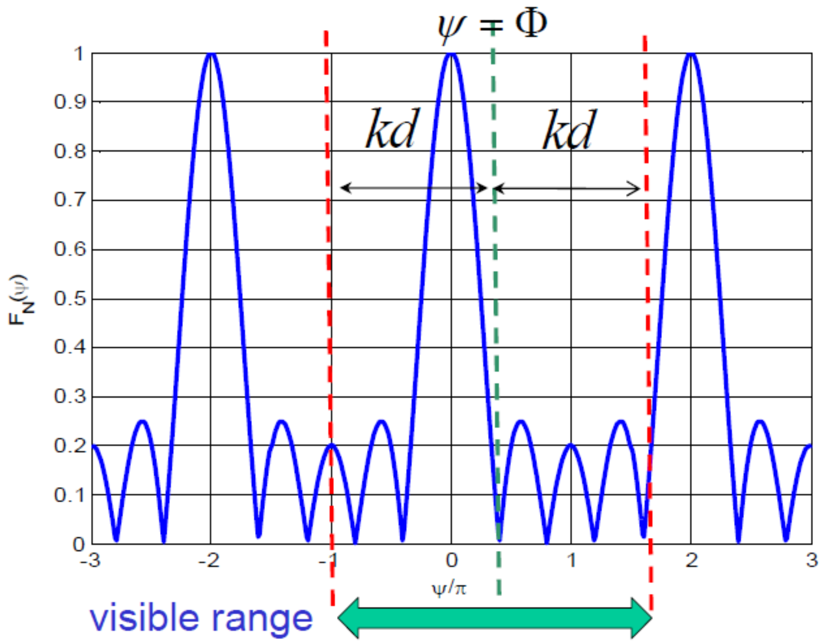
$$AF(\theta,\phi)=\sum_{n=1}^{N}\exp[+j(n-1)\psi], \quad \psi\equiv kd(\hat{r}\cdot\hat{u})+\Phi$$

DET

Politecnico di Torino 1859

OPTIMAL PLACEMENT OF ANTENNAS IN AN ARRAY - IV

1D Equispaced Uniform Phased Arrays



The AF is defined over the entire real axis, and it is periodic.
The variable ψ is limited to a finite interval in the visible range, defined as:

$$\psi^{visible} \in [-kd + \Phi, +kd + \Phi]$$

Everything at ψ locations outside visible range (e.g. zeros, lobes), is not present in radiation pattern!

$$F_N(\psi) = \frac{1}{N} \left| \frac{\sin(N\frac{\psi}{2})}{\sin\frac{\psi}{2}} \right|$$

$$\psi = kd\hat{r} \cdot \hat{u} + \Phi = kd\cos\alpha + \Phi$$

➡ Closed form design equations available (degrees of freedom are ***N***, ***d*/λ** and **Φ**)

DET

Politecnico di Torino 1859

Optimal placement of antennas in an array - V

	Side Lobes Level Amplitude	Beam Forming Network complexity
Uniform Amplitude	☹️	🙂
Non-Uniform Amplitude	🙂	☹️
Sparse Array	🙂	🙂

For sparse array, *there is no (general) analytical design procedure.* ☹️

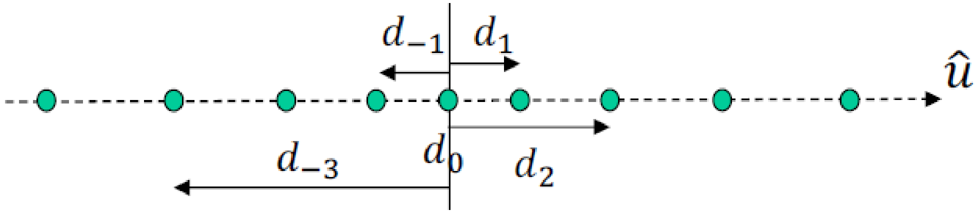
⬇

Design based on global optimization

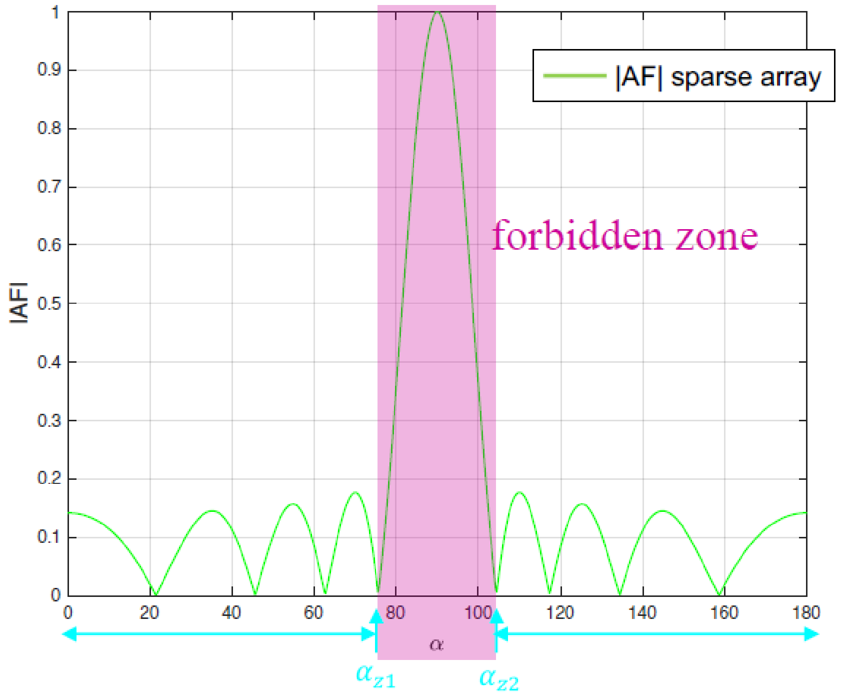
Optimal placement of antennas in an array - VI

Application of Genetic Algorithm to the design of a sparse array

Goal = The minimization of the maximum SLL.



The N array elements are not *equispaced*, but *symmetrical* with respect to center.



- *Cost value:* Maximum of the (normalized) AF out of the main beam, i.e., in the interval:

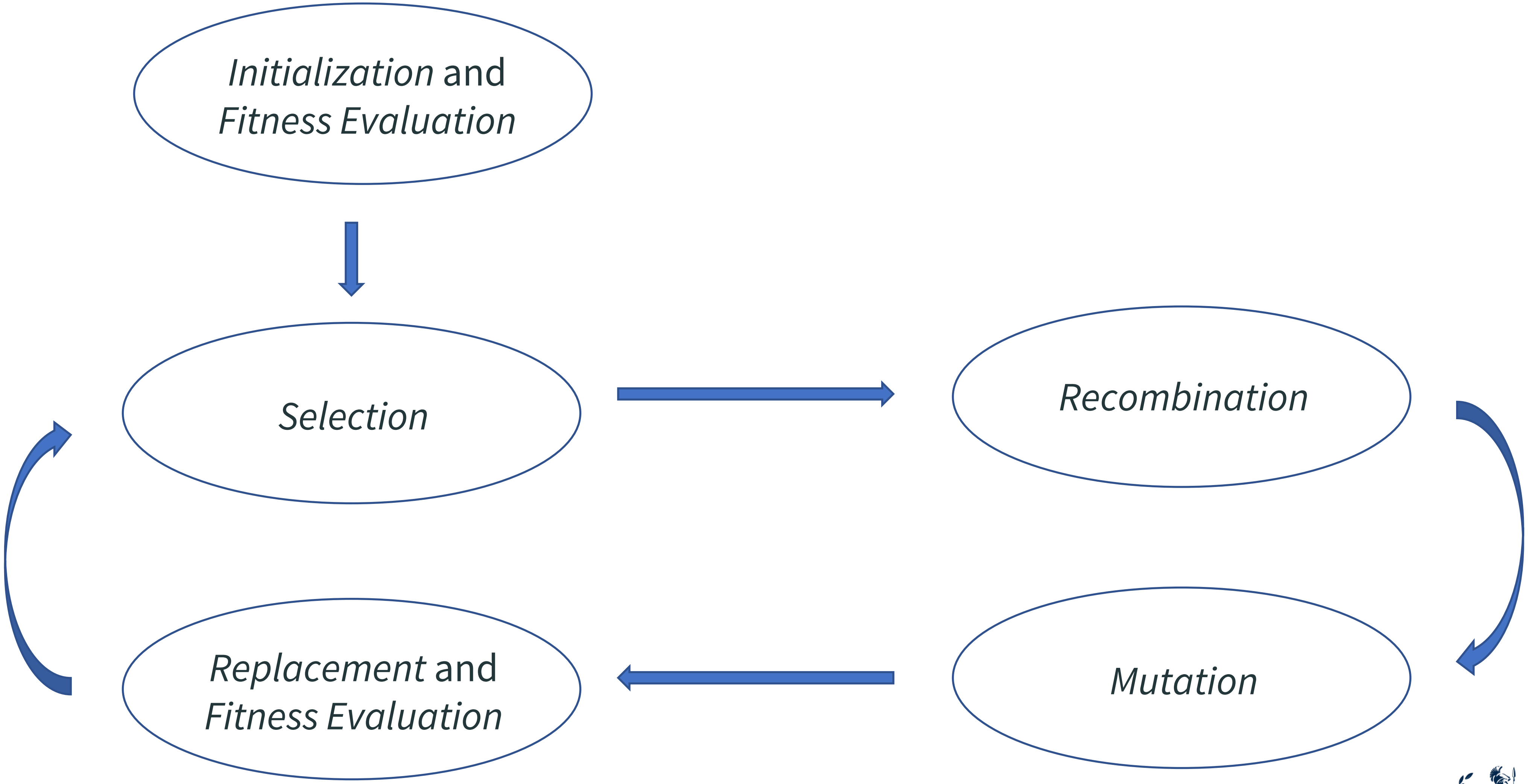
$$max\{AF(\alpha)\}, \quad \alpha \in [0°, \alpha_{z1}] \cup [\alpha_{z2} 180°]$$

- *Independent variables*: position of the N array elements.

$$\frac{d_n}{\lambda} \text{ as } \frac{d_n}{\lambda} = n\frac{d_{min}}{\lambda} + \frac{d_n^{GA}}{\lambda}$$

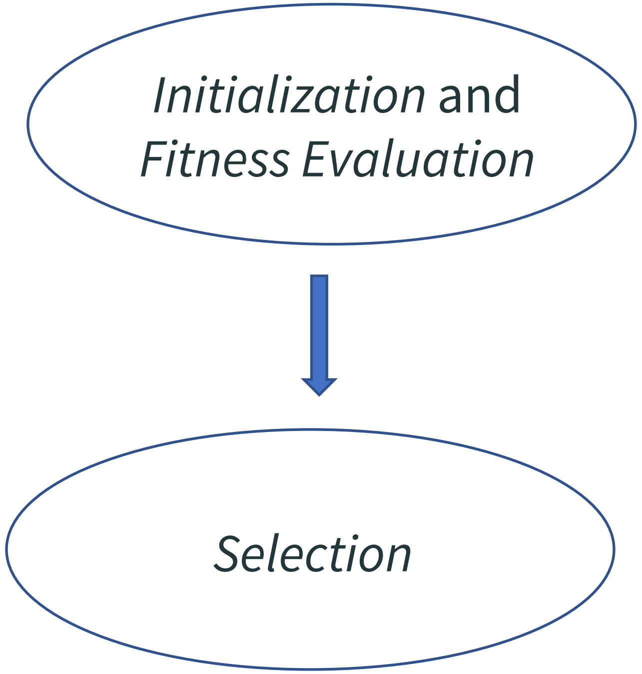
Optimal placement of antennas in an array - VII

Application of Genetic Algorithm to the design of a sparse array



OPTIMAL PLACEMENT OF ANTENNAS IN AN ARRAY - VIII

Application of Genetic Algorithm to the design of a sparse array

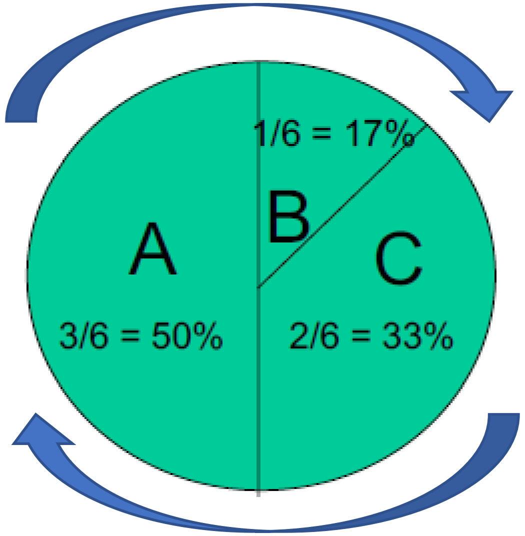


- Population size ($N_{population}$ = 50) is set, and the population individual are defined.
  Each individual is composed of a set of $\frac{d_n^{GA}}{\lambda}$, which allows to determine the correspondent cost and fitness values.

- A fitness proportionate selection method is adopted, i.e., *roulette-wheel selection.*

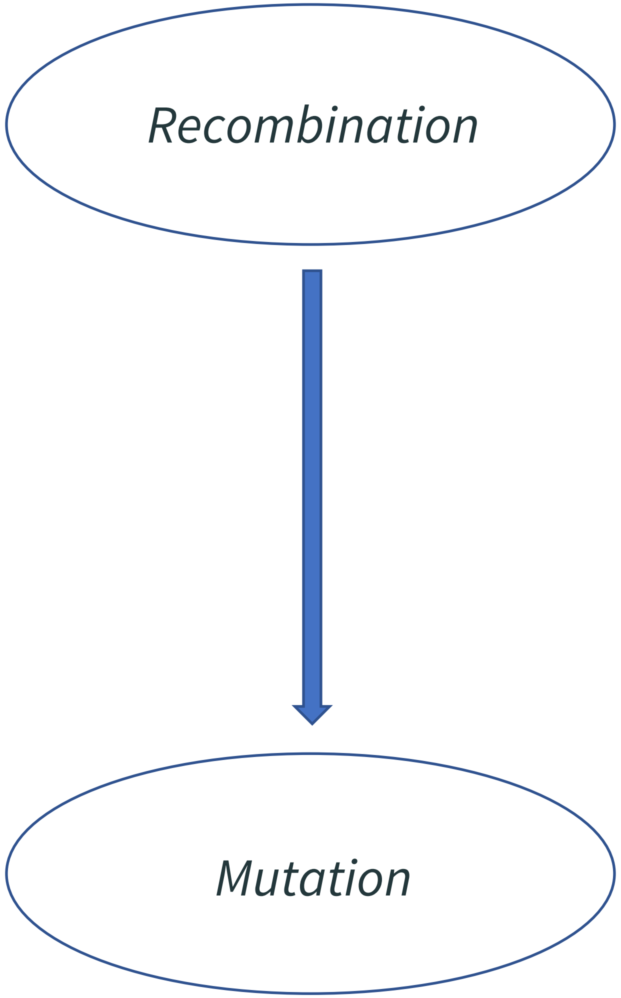
- Each individual in the population is assigned a roulette wheel slot sized in proportion to its fitness.

- *Spin* the wheel *n* times to select *n* individuals.



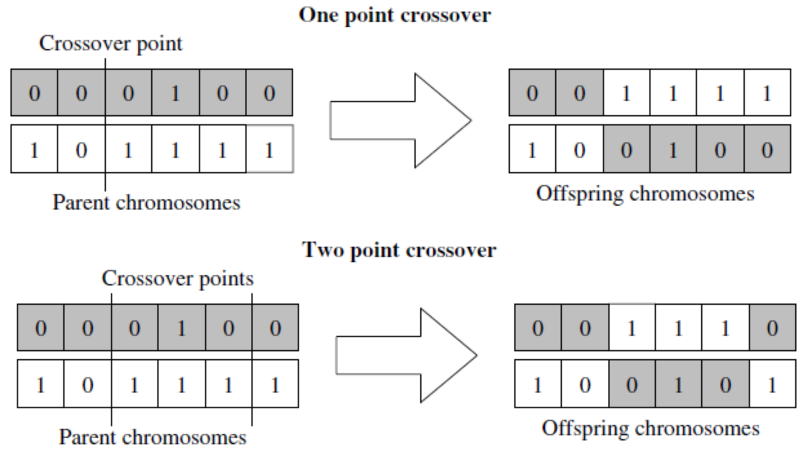
Optimal placement of antennas in an array - IX

Application of Genetic Algorithm to the design of a sparse array



- The selected individuals from the mating pool are recombined to create new, hopefully better, offspring. This is done considering a certain crossover probability (P_{cross} = 0.5).

  The exploited approaches to recombine individuals are:

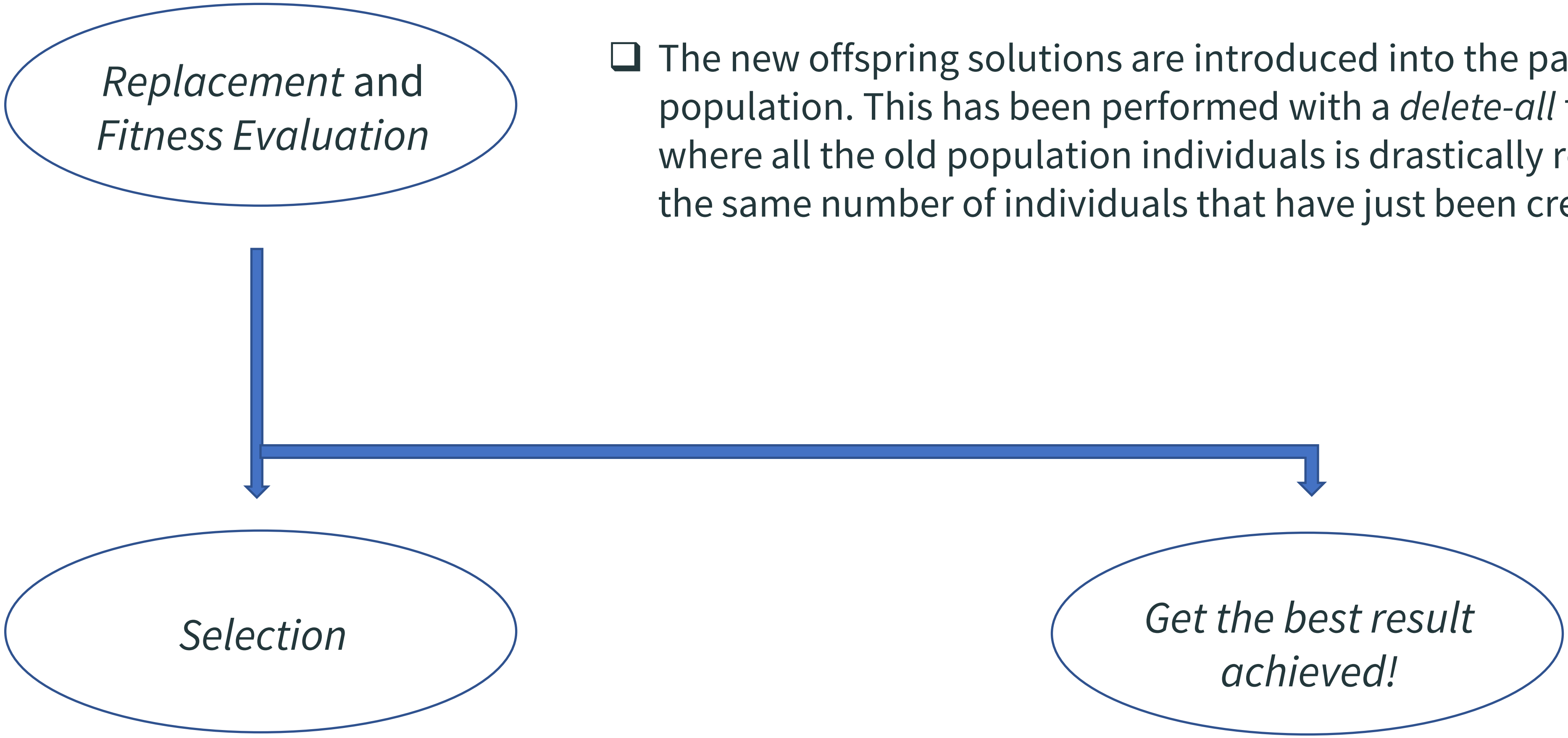


- Depending on the mutation probability (equal to $P_{mutation}$ = 0.1), the chromosome's elements are randomly changed.

Optimal placement of antennas in an array - X

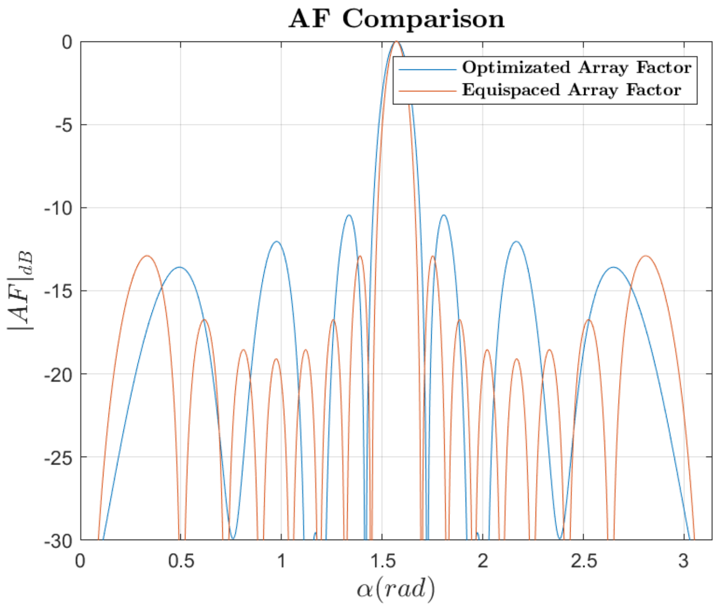
Application of Genetic Algorithm to the design of a sparse array

- ❑ The new offspring solutions are introduced into the parental population. This has been performed with a *delete-all* technique, where all the old population individuals is drastically replaced with the same number of individuals that have just been created.



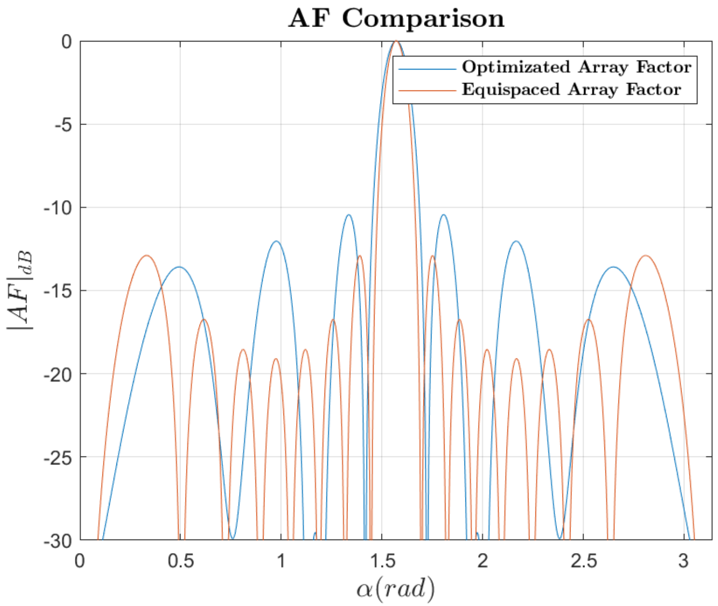
OPTIMAL PLACEMENT OF ANTENNAS IN AN ARRAY - XI

Optimization problem - Results



- One-point crossover.

$SLL_{red} = 4$ dB



- Two-point crossover.

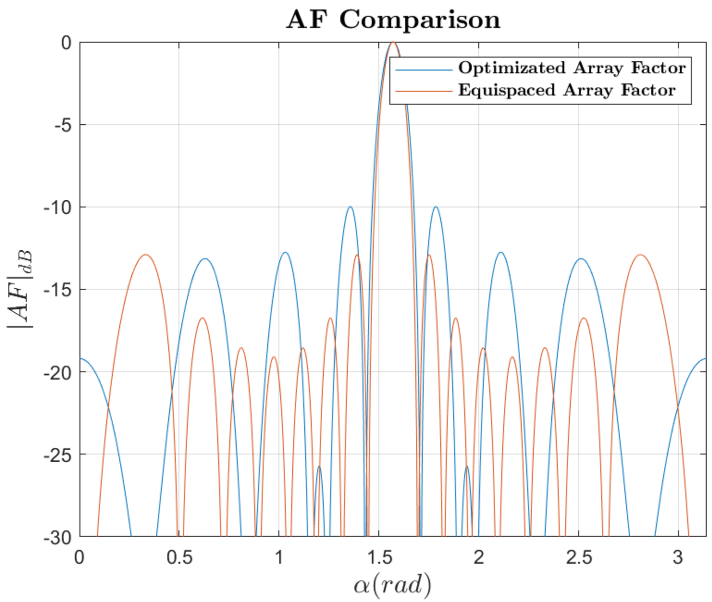
$SLL_{red} = 4$ dB

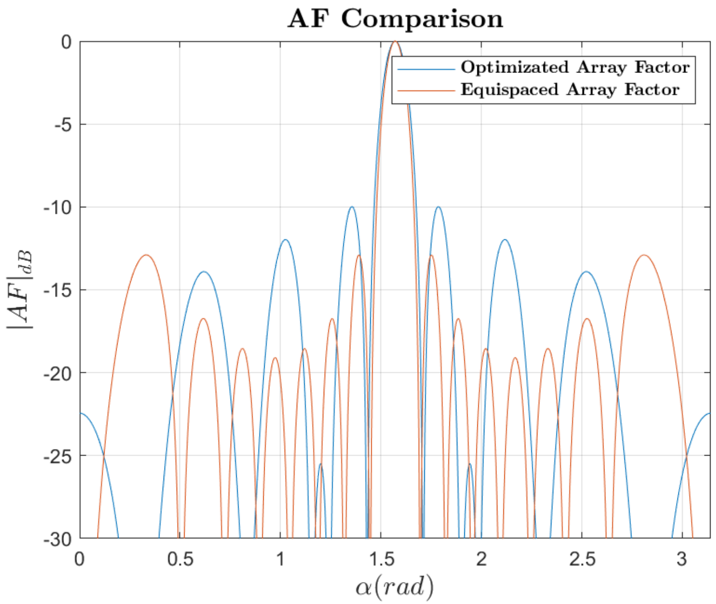
- Number of array elements N = 9.
- $\dfrac{d_{min}}{\lambda} = 0.6$
- $N_{Trial} = 50$

Politecnico di Torino 1859

Optimal placement of antennas in an array - XI

Optimization problem - Results





- ❑ Number of array elements N = 9.
- ❑ $\frac{d_{min}}{\lambda} = 0.65$
- ❑ $N_{Trial} = 50$

- ❑ One-point crossover.

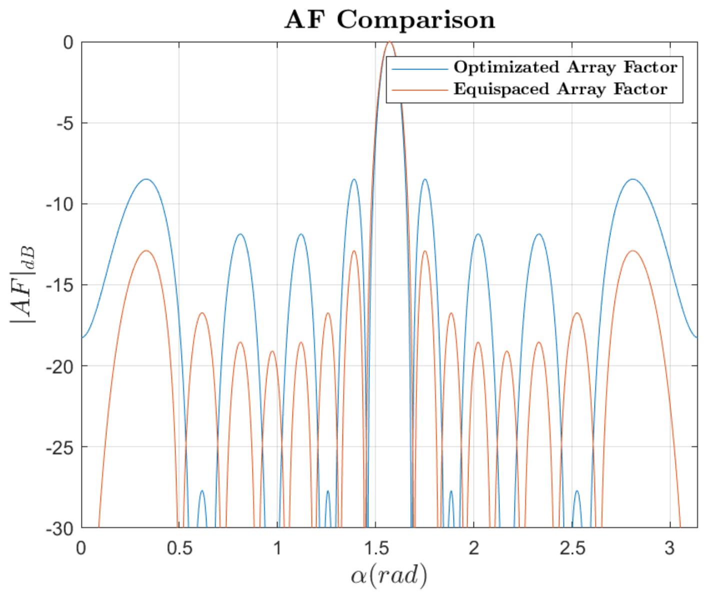
$SLL_{red} = 2.5$ dB

- ❑ Two-point crossover.

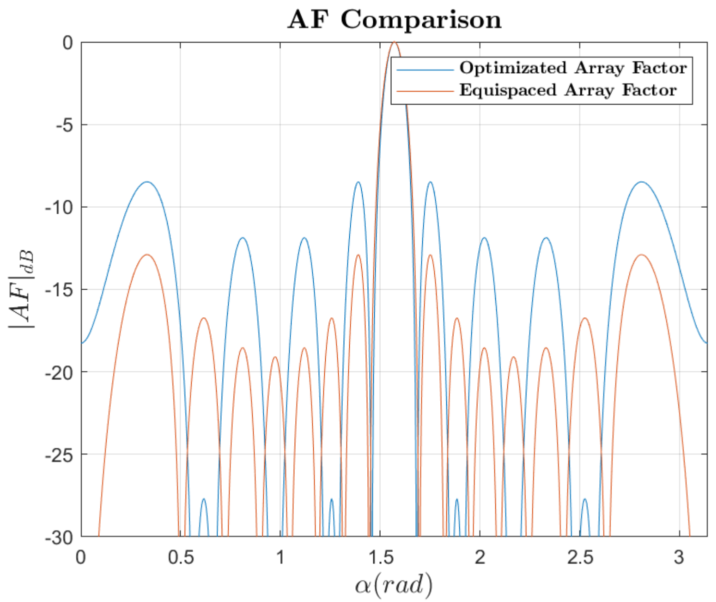
$SLL_{red} = 3$ dB

Optimal placement of antennas in an array - XI

Optimization problem - Results



- ❑ One-point crossover.



- ❑ Two-point crossover.

- ❑ Number of array elements N = 9.
- ❑ $\frac{d_{min}}{\lambda}$ = 1 – 1/N
- ❑ N_{Trial} = 50

No SLL_{red} !
Uniform array coincide with the optimum.

PART III: Application to QUBO problems

QUBO FORMULATION

The acronym stands for **Quadratic Unconstrained Binary Optimization**:

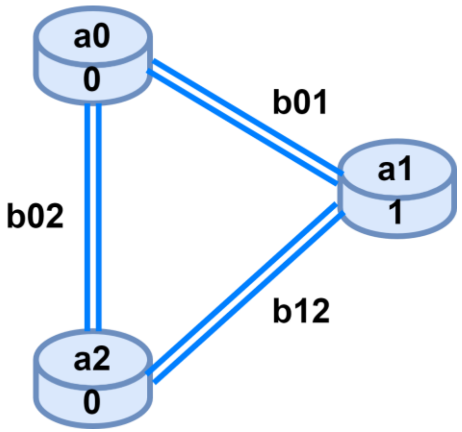
- **Quadratic** refers on the highest power applied on variables
- **Unconstrained** means that no constraints are applied to a variable
- **Binary** because the involved variables can assume only 0 or 1 values
- **Optimization** because this model is used to minimize the obtained objective functions

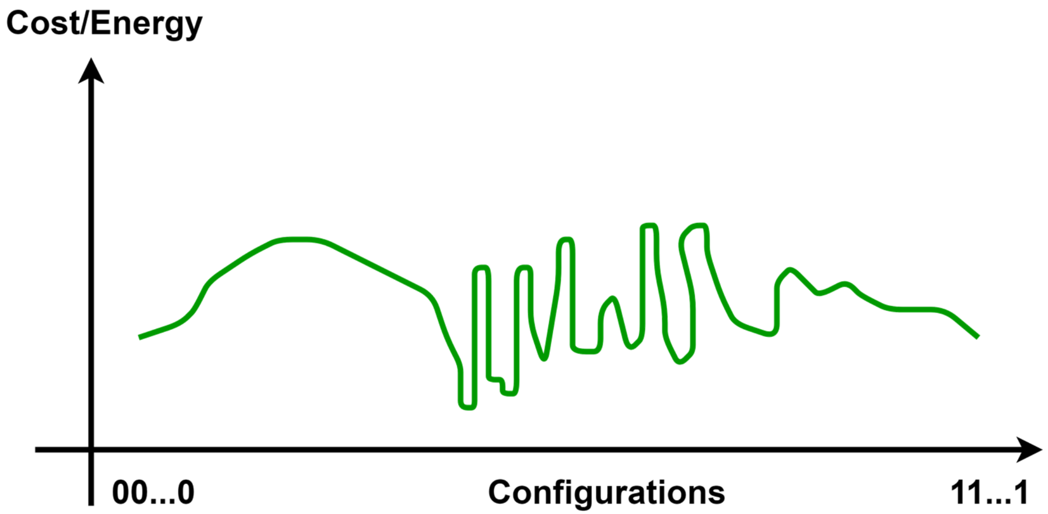
$$y = x^t Q x$$

Vector of binary variables (x)

QUBO matrix (Q)

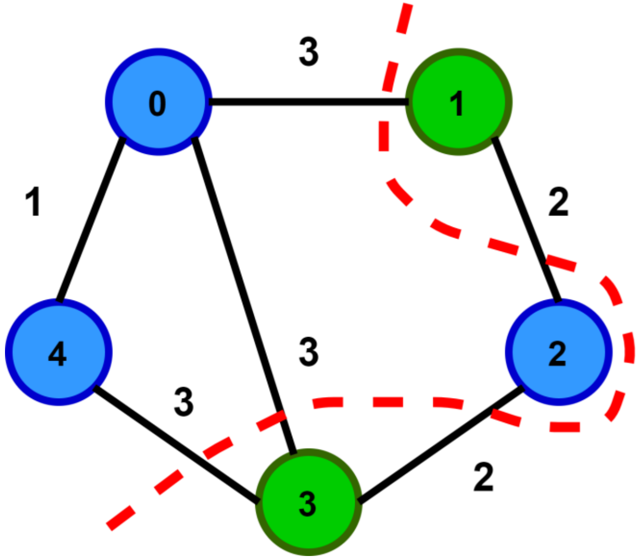
Despite from the appearence, many types of constraints can be formulated with the **QUBO** model by introducing **quadratic penalties** to the objective function.



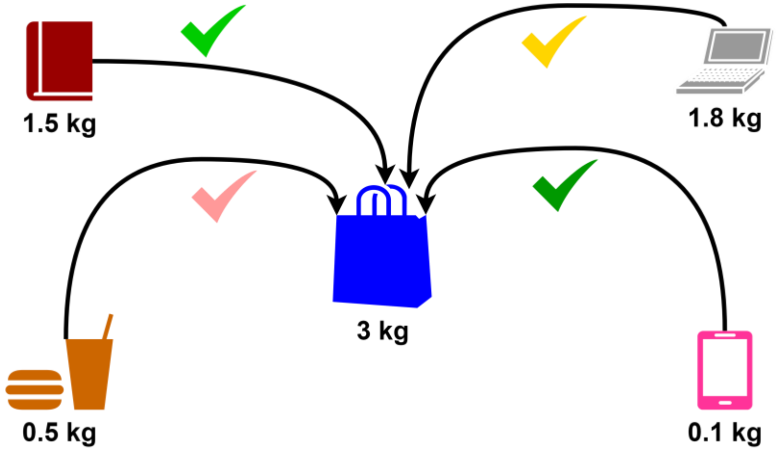


Benchmark problems considered

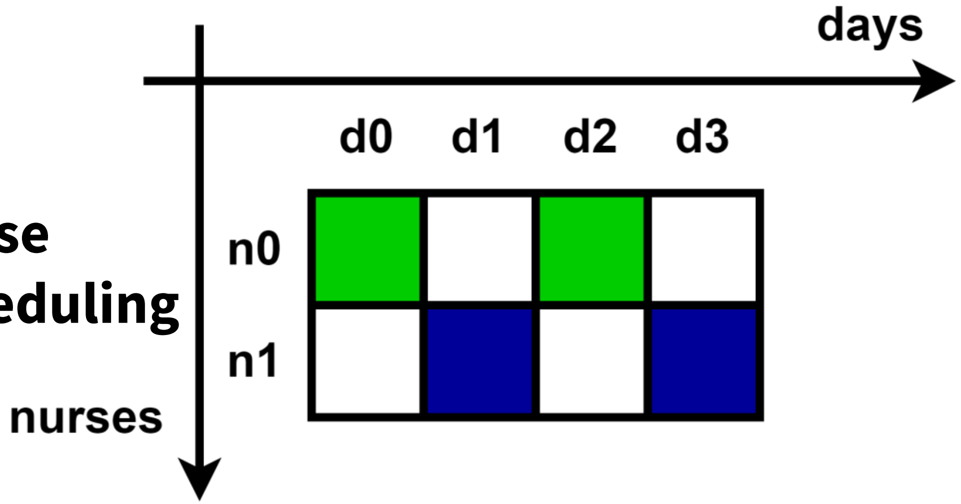
Maxcut



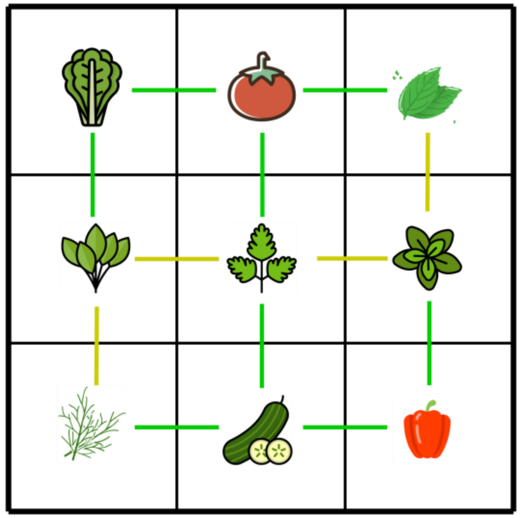
Knapsack



Nurse scheduling



Garden optimization

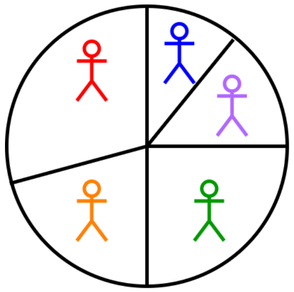


Cost function shapes different from each other

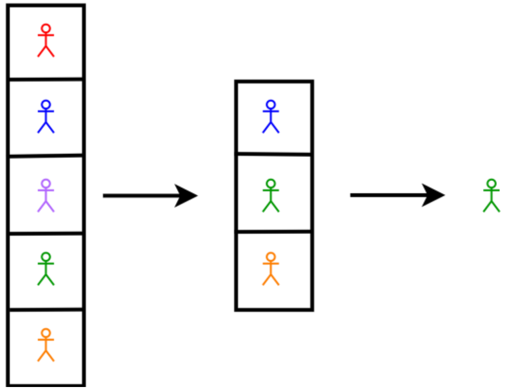
Selection, Crossover and replacement approaches supported

Selection:

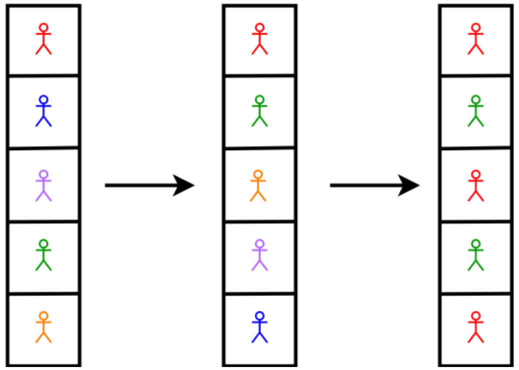
- Roulette wheel selection



- Tournament selection

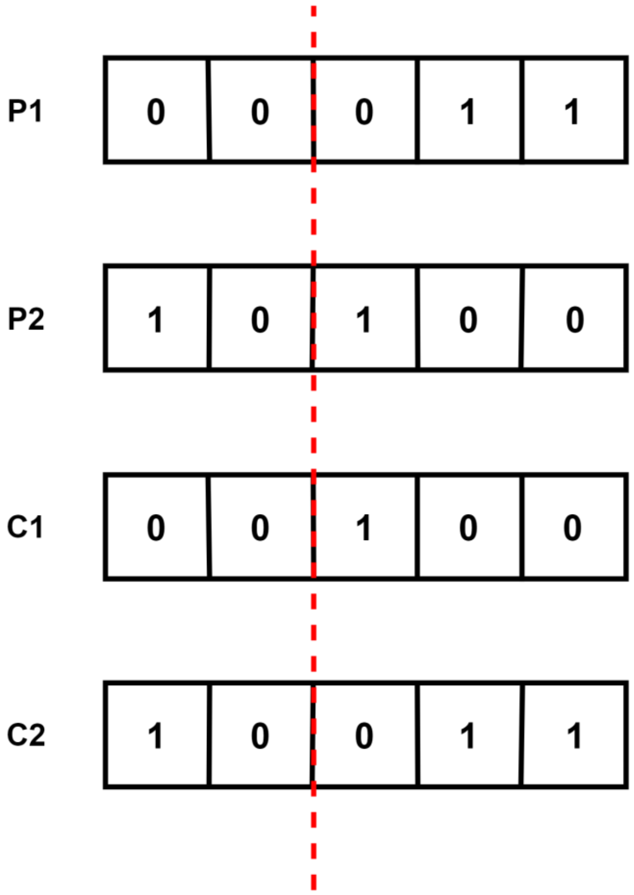


- Truncation selection



Crossover:

- 1-point
- 2-point
- Uniform

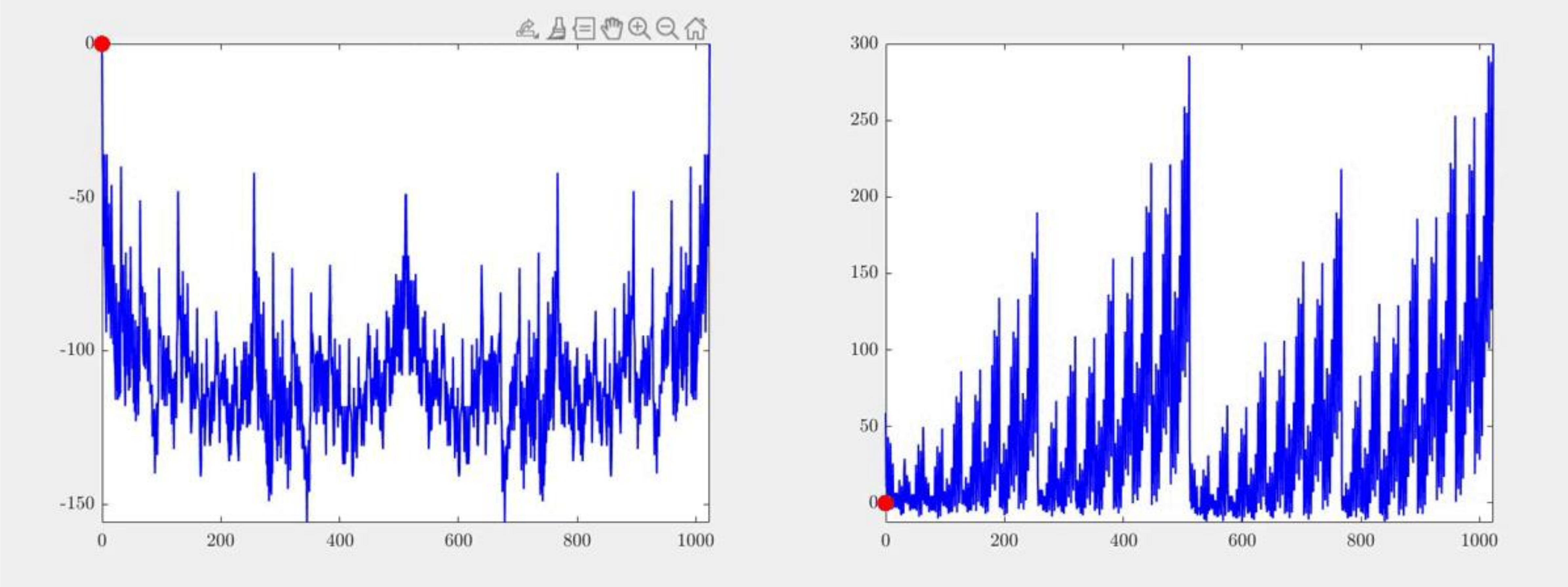


Replacement:

- Delete-All

Politecnico di Torino

Solution space exploration – Maxcut and Knapsack



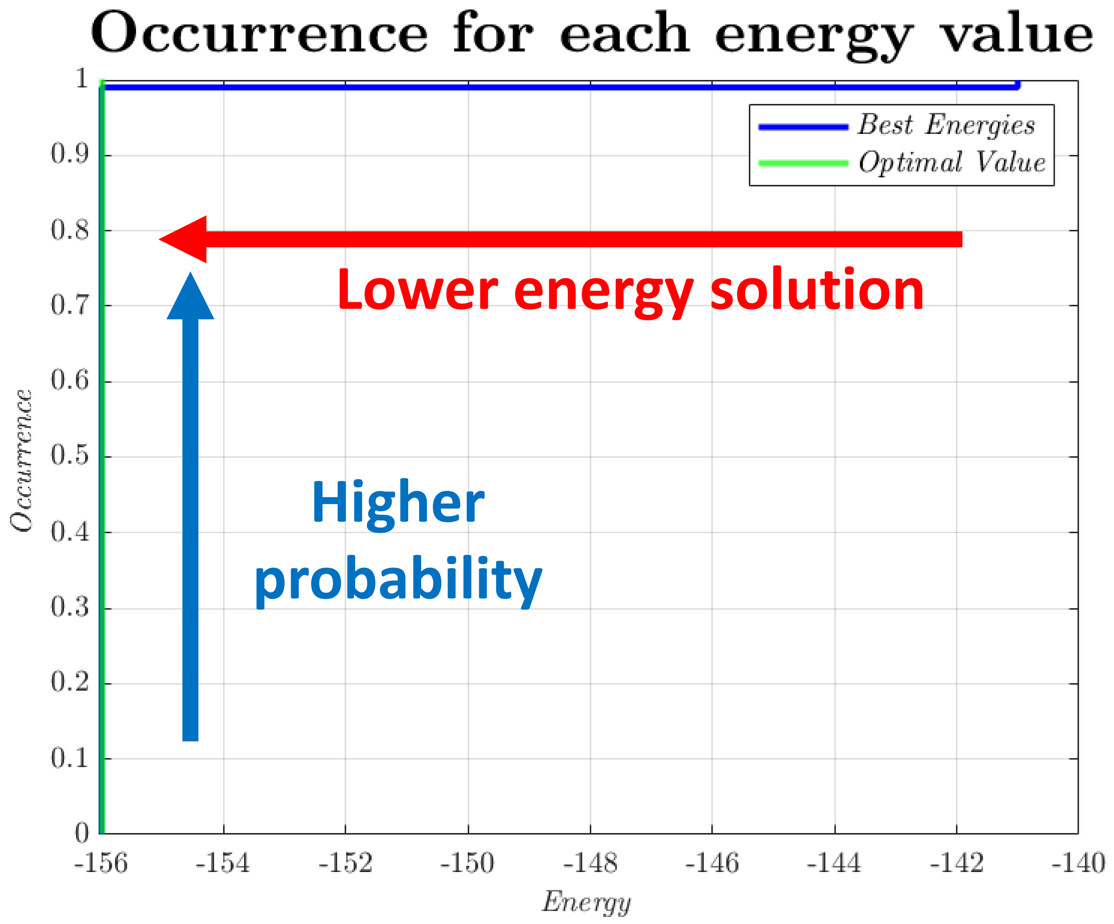
Maxcut 10 nodes: tournament selection (s=2), 2-point crossover, mutation (p_m=0.2), 4 elements in the population, 10 iterations

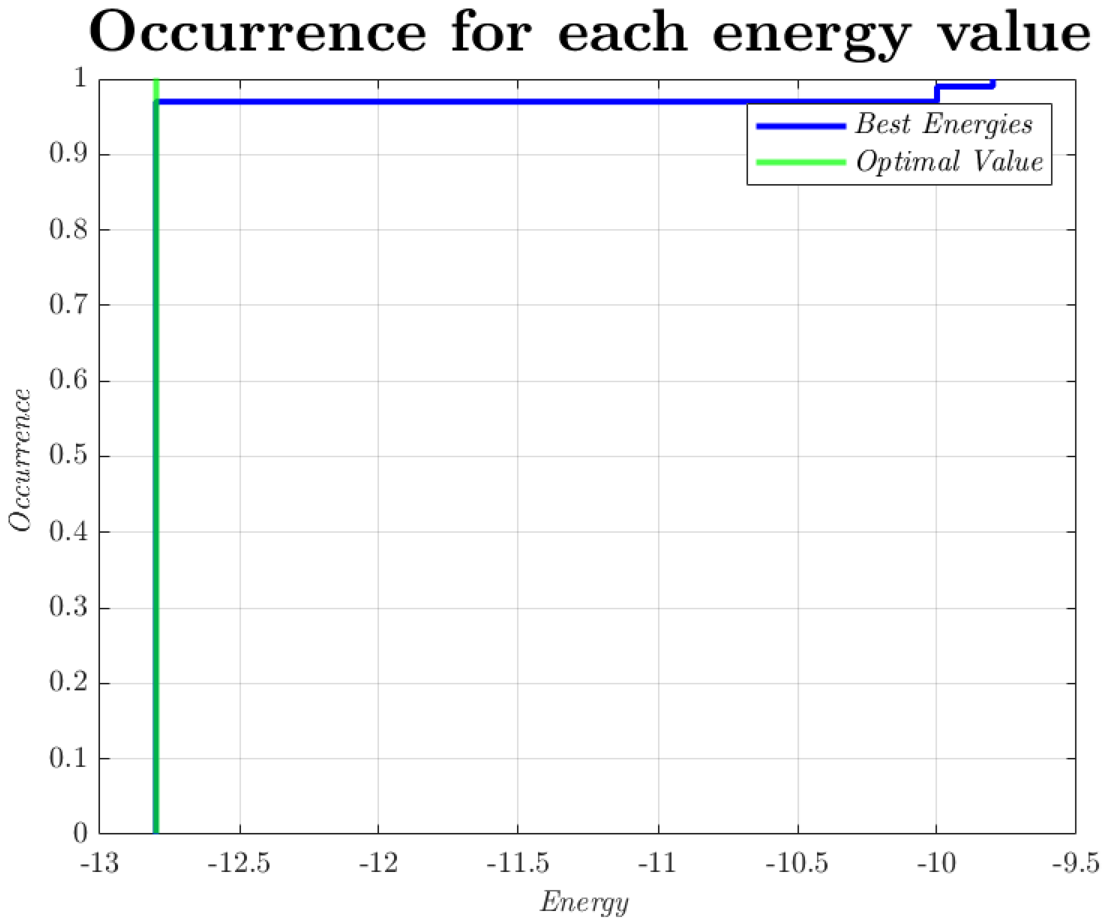
Knapsack 10 variables: roulette wheel selection, uniform crossover (p_c = 0.5), mutation (p_m=0.2), 4 elements in the population, 10 iterations

DET

Politecnico di Torino
1859

CUMULATIVE DISTRIBUTION: MAXCUT AND KNAPSACK



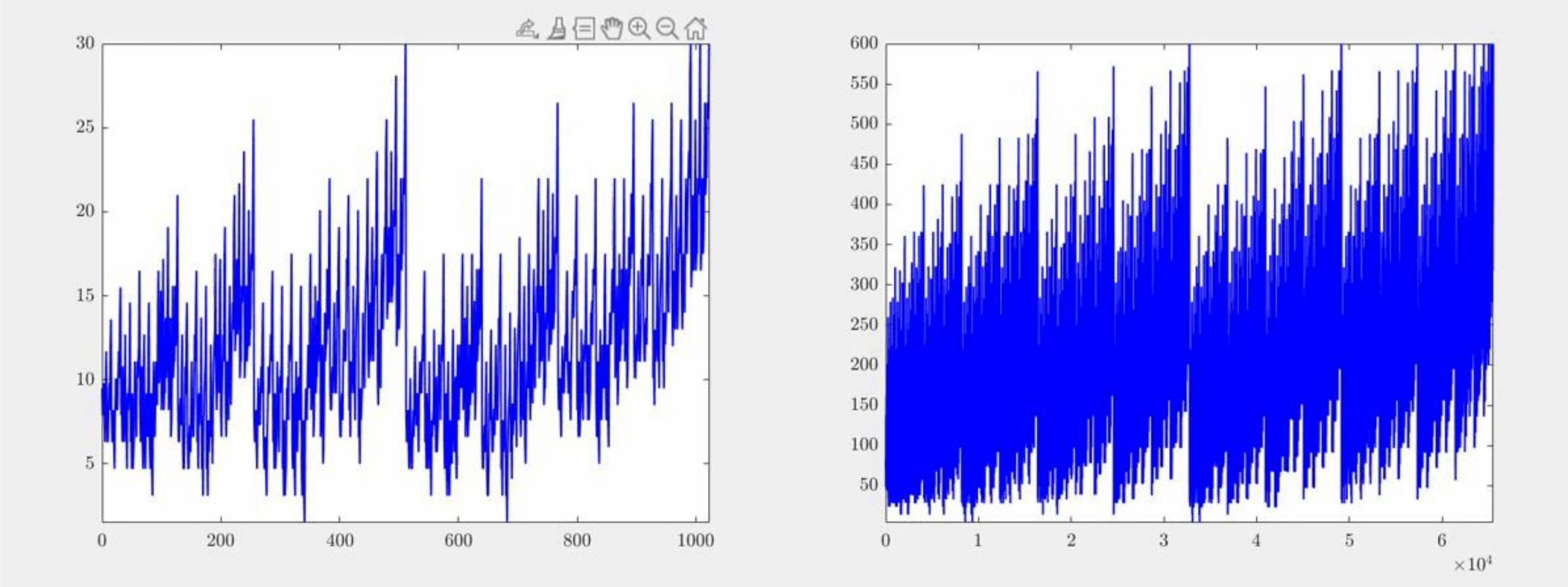


Maxcut 10 nodes: tournament selection (s=2), 2-point crossover, mutation (p_m=0.2), 4 element in the population, 10 iteration

Knapsack 10 variables: roulette wheel selection, uniform crossover (p_c = 0.5), mutation (p_m=0.2), 4 element in the population, 10 iteration

Politecnico di Torino 1859

Solution space exploration – Nurse scheduling and Garden



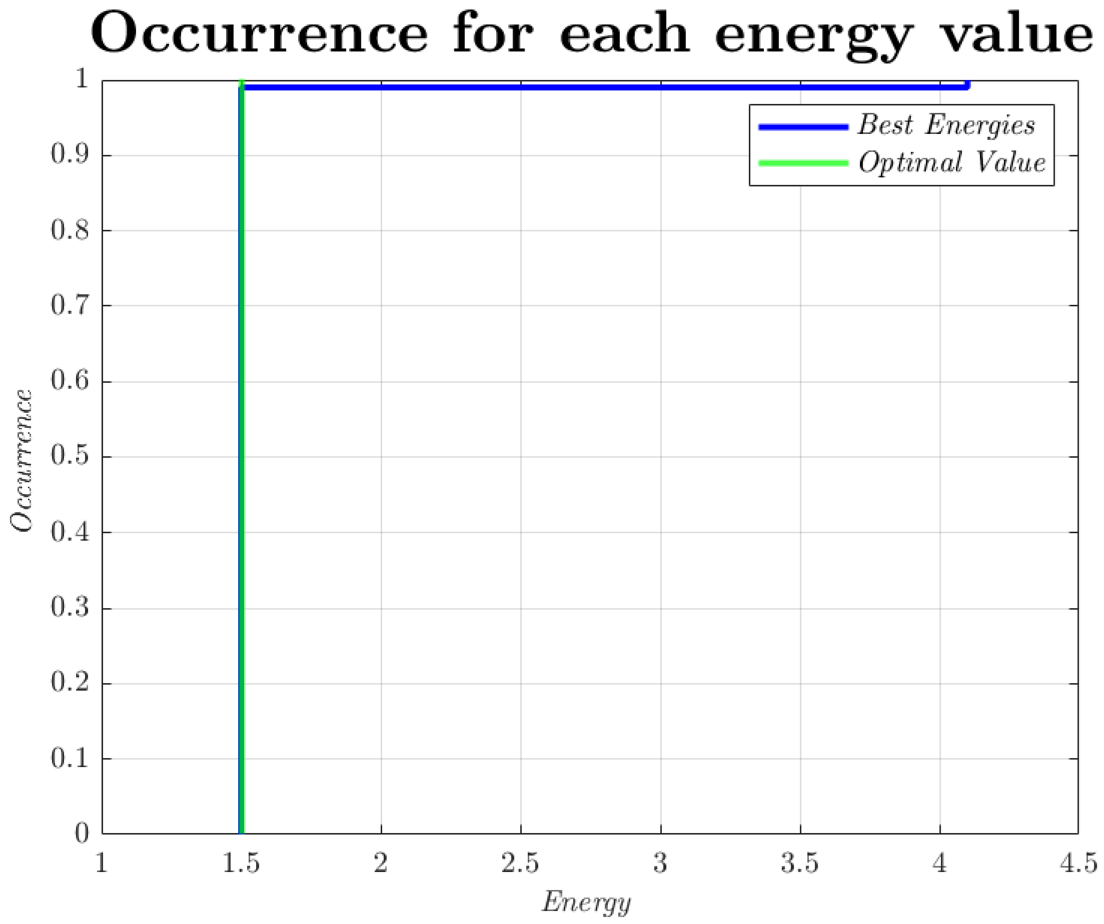
Nurse scheduling 5 days and 2 nurses: tournament selection (s=2), 2-point crossover, mutation (p_m=0.2), 4 elements in the population, 10 iterations

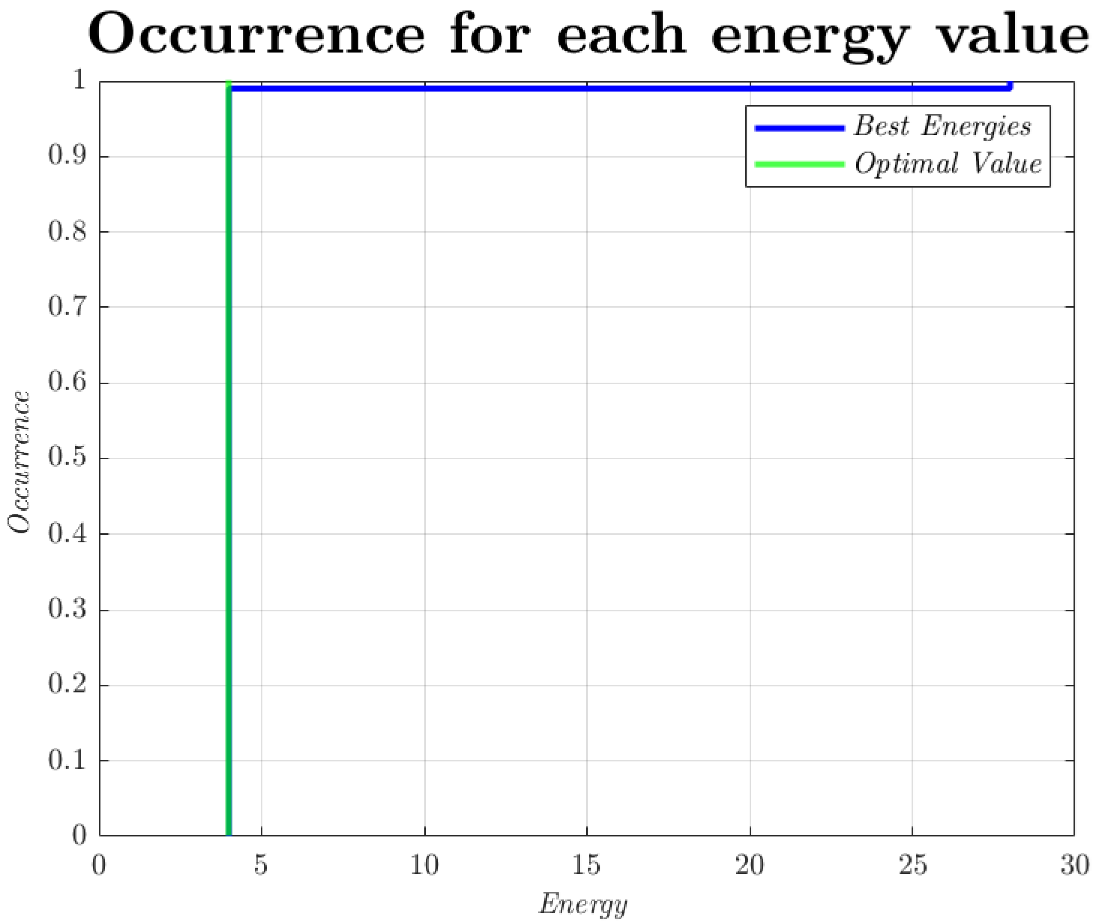
Garden Optimization 16 variables: truncation selection (s=2), uniform crossover (p_c = 0.7), mutation (p_m=0.2), 4 elements in the population, 20 iterations

DET

Politecnico di Torino
1859

Cumulative distribution: Nurse scheduling and Garden





Nurse scheduling 5 days and 2 nurses: tournament selection (s=2), 2-point crossover, mutation (p_m=0.2), 4 element in the population, 10 iteration

Garden Optimization 16 variables: truncation selection (s=2), uniform crossover (p_c = 0.7), mutation (p_m=0.2), 4 element in the population, 20 iteration

DET

Politecnico di Torino 1859

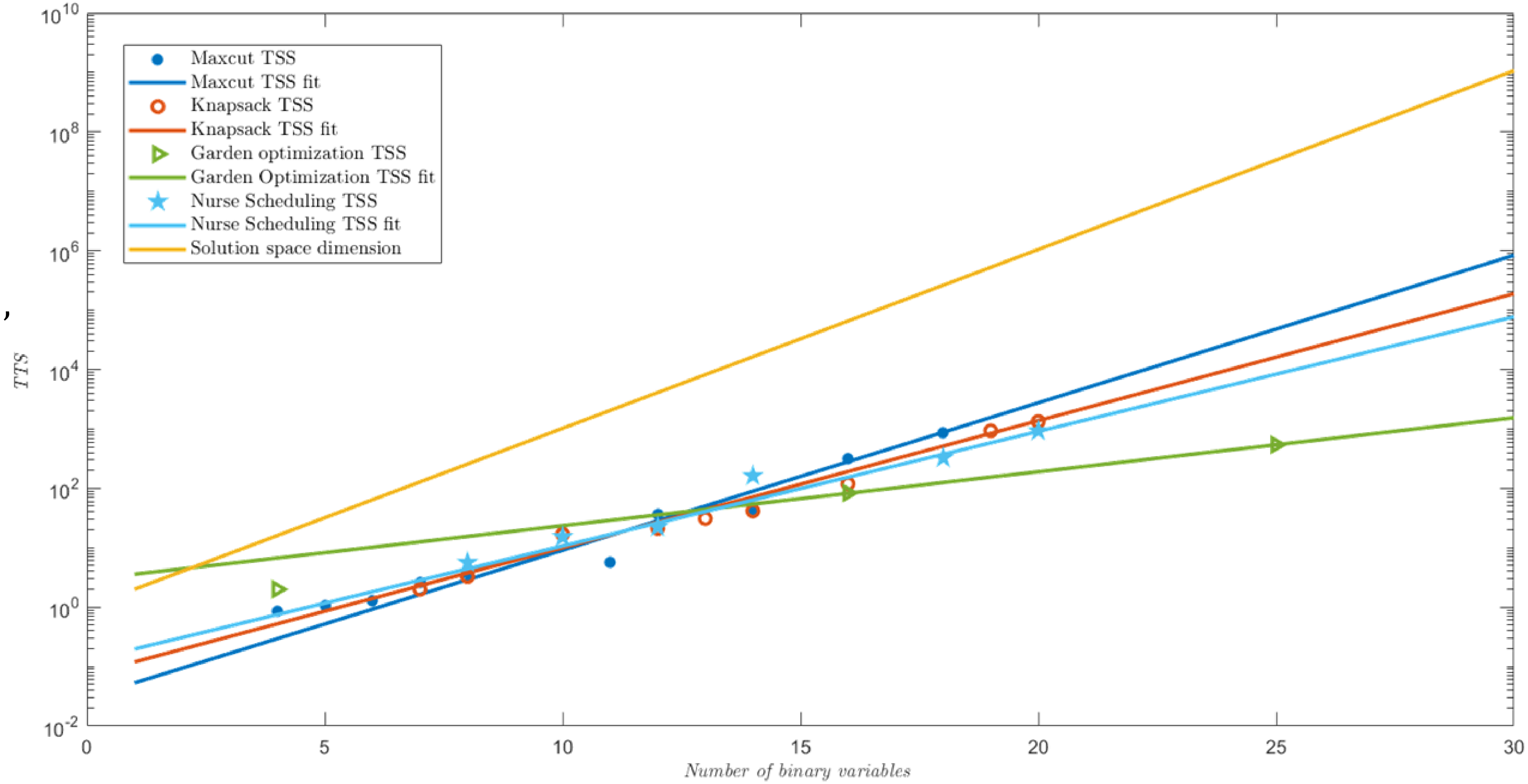
Scaling

TTS (time-to-solution) permits to estimate the number of iterations required for obtaining a success probability of **95%:**

$$TTS(t_f) = t_f \frac{\ln(1-0.95)}{\ln(1-p_s(t_f))},$$

where p_s is the success probability obtained by executing t_f iterations of the algorithm.

[Demostration of a Scaling Advantage for a Quantum Annealer over Simulated Annealing](), Tameem Albash and Daniel A. Lidar

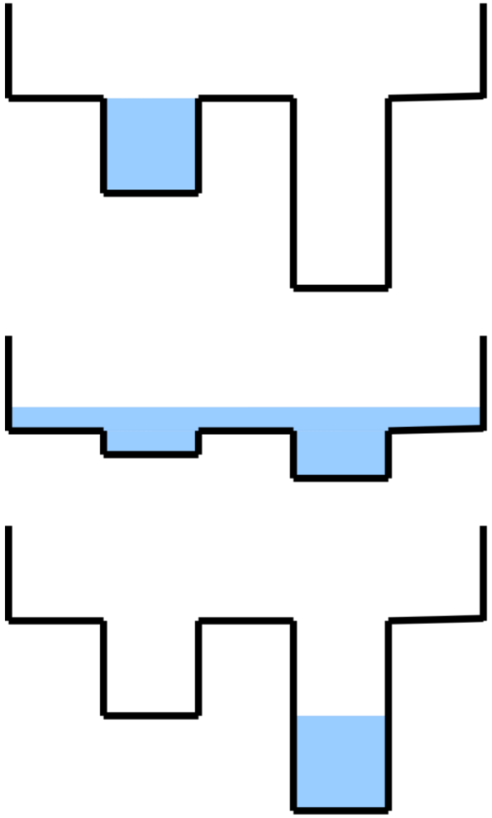


Politecnico di Torino

PART IV: Conclusions and future perspectives

Future Perspectives: Quantum-classical genetic algorithm

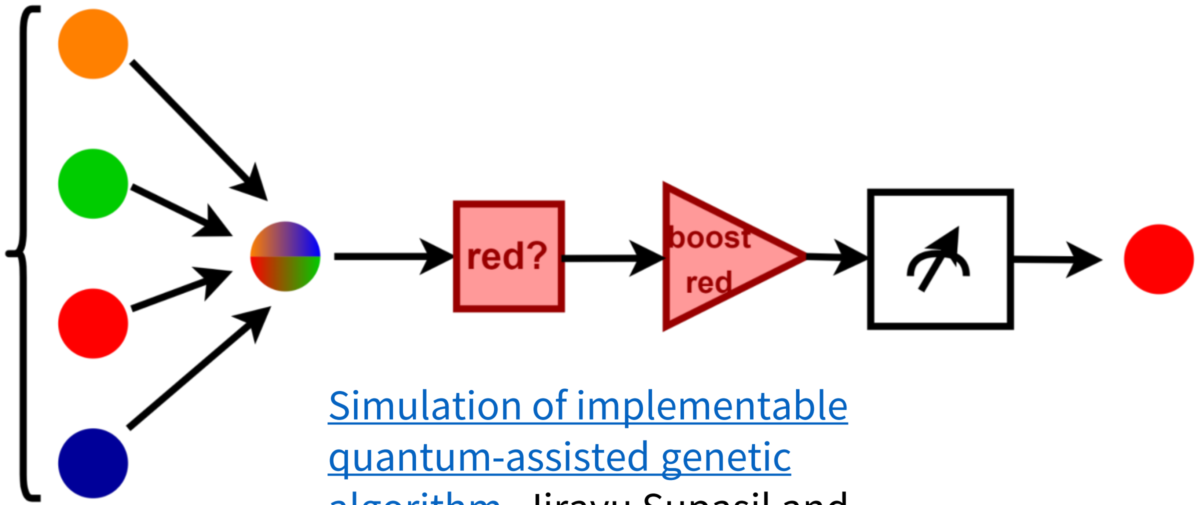
QUANTUM ASSISTED GENETIC ALGORITHM



Reverse annealing for mutation

Quantum-Assisted genetic algorithm, King J. and Mohseni M. and Bernoudy W. and Fréchette A. and Sadeghi H. and Isakov S. V. and Neven H. and Amin M. H.

Grover Search instead of crossover and mutation

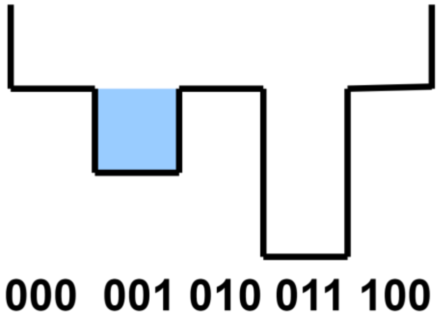


Simulation of implementable quantum-assisted genetic algorithm, Jirayu Supasil and Poramet Pathumsoot and Sujin Suwanna.

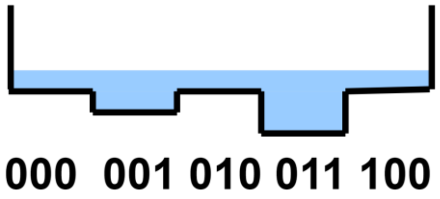
DET

Politecnico di Torino
1859

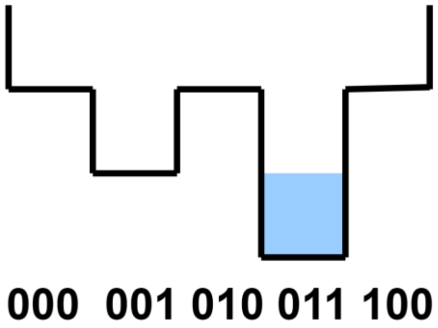
Future Perspectives: Reverse Quantum Annealing



The starting classical solution is obtained after crossover operator



By applying a field, a quantum superposition of states it obtained



By reducing the applied field, the system collapses in a new classical state, which is the new solution. This is useful as it allows to escape from local minima, like mutation does.

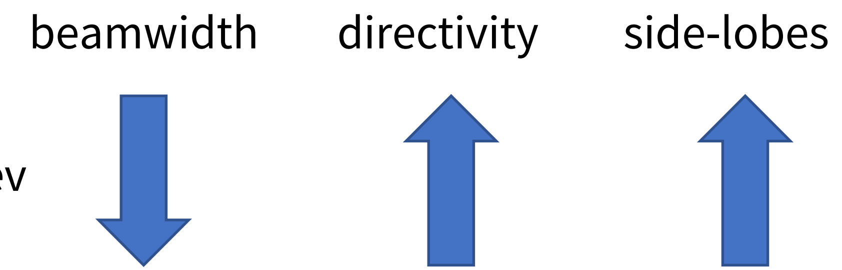
PART V: Bibliography

Bibliography

Ridwan et al. "Design of Non-Uniform Antenna Arrays Using Genetic Algorithm"

- Antenna arrays allow to obtain high directivity, narrow beamwidth and low side-lobes

	Uniform antenna array	Non-uniform antenna array (e.g. binomial, Dolph-Chebyshev)
Uniform inter-element spacing	✓	✓
Uniform Amplitude	✓	✗

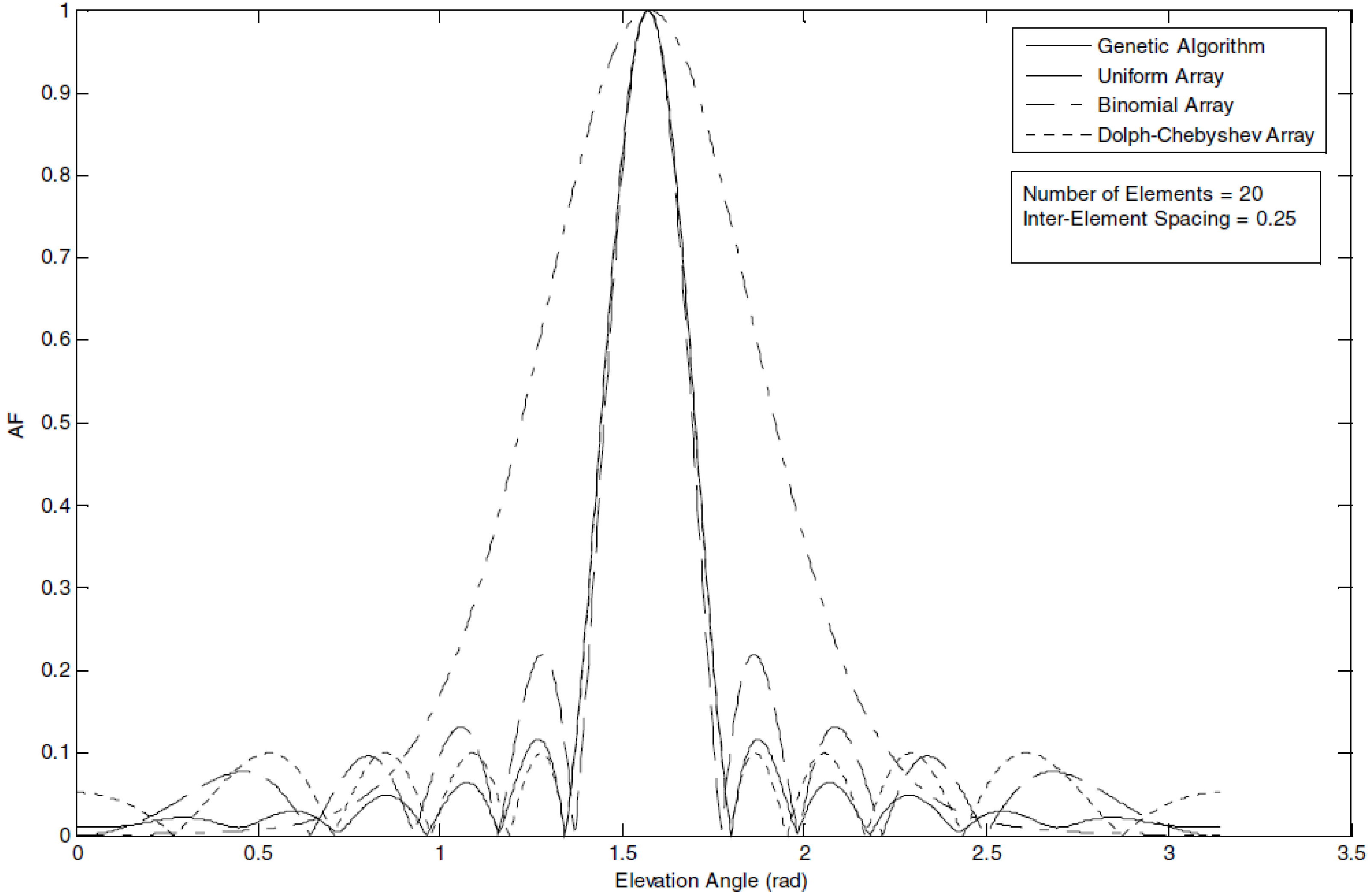


- Uniform array
- Dolph-Chebyshev
- Binomial arrays

- Goal: design a non-uniform array that approximates the beamwidth of a uniform array and having smaller side-lobe level than the Dolph-Chebyshev array

LITERATURE REVIEW

- GA is used to find the excitation amplitudes that allow to optimize the antenna array design



Politecnico di Torino 1859

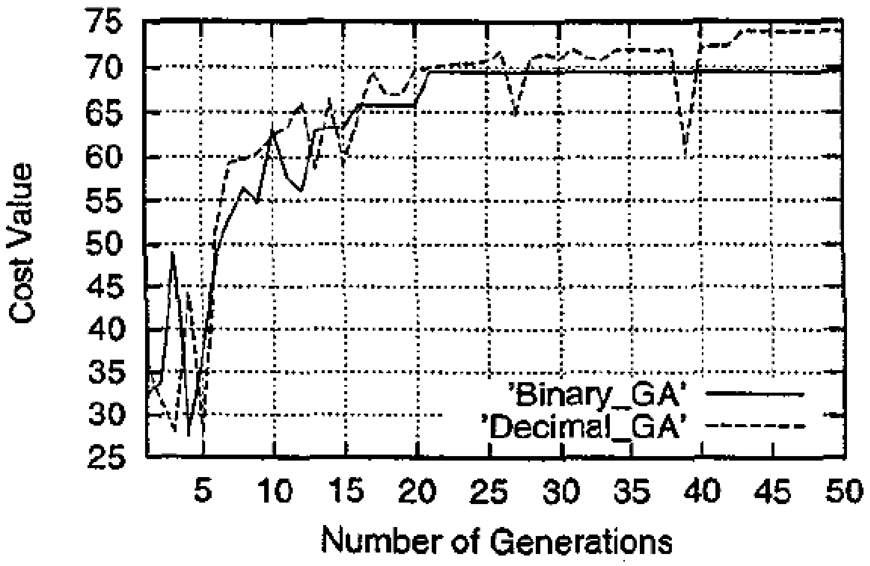
LITERATURE REVIEW

Lee et al. "Genetic algorithm using real parameters for array antenna design optimisation."

- GA is applied to optimize the design of a realistic antenna
- Continuous genetic algorithm: instead of using binary strings, individuals are represented by means of real parameters
  - No need to encode real parameters into binary values: more efficient code
  - In binary GA, precision is influenced by the number of bits used to encode parameters; with real numbers this issue does not occur
- Crossover: a crossover factor F is employed; its choice is very important, as it "determines how well the search space is being searched".
  - $C_1 = (1 - F)P_1 + FP_2$
  - $C_2 = (1 - F)P_2 + FP_1$
- In binary GA, lower probability of a high crossover point, thus significant bits are not changed. However, “with an appropriate value for F, the probability of crossover at more significant bits is increased. This results in a more rigorous search of the entire problem space”

BIBLIOGRAPHY

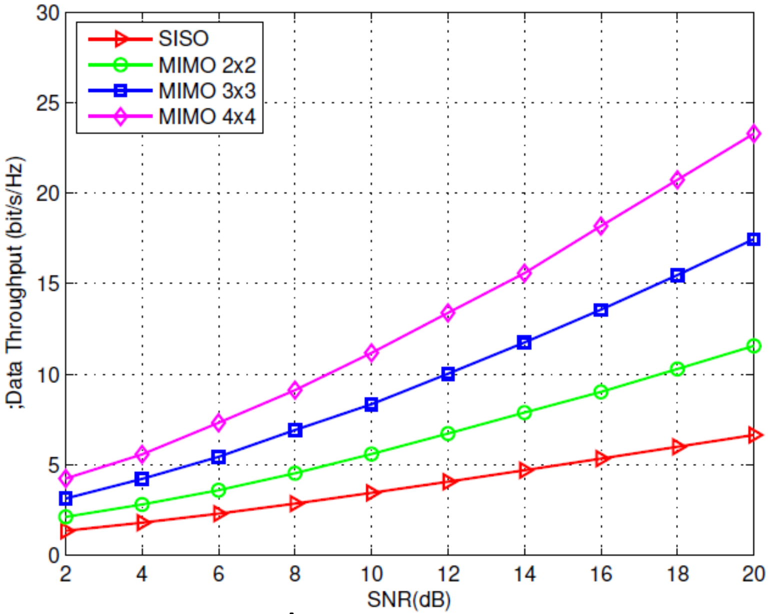
- Mutation
  - In binary GA: randomly flipping a bit
  - In continuous GA: randomly altering the value of parameters
    - $P_i = P_i \pm F_{mut}R$, with $R = P_{max} - P_{min}$
- Replacement: Elitism Roulette Wheel Algorithm, i.e. fittest top 10% individuals are selected for the new population directly; remaining 90% is chosen by using the roulette wheel algorithm
- Final plot:



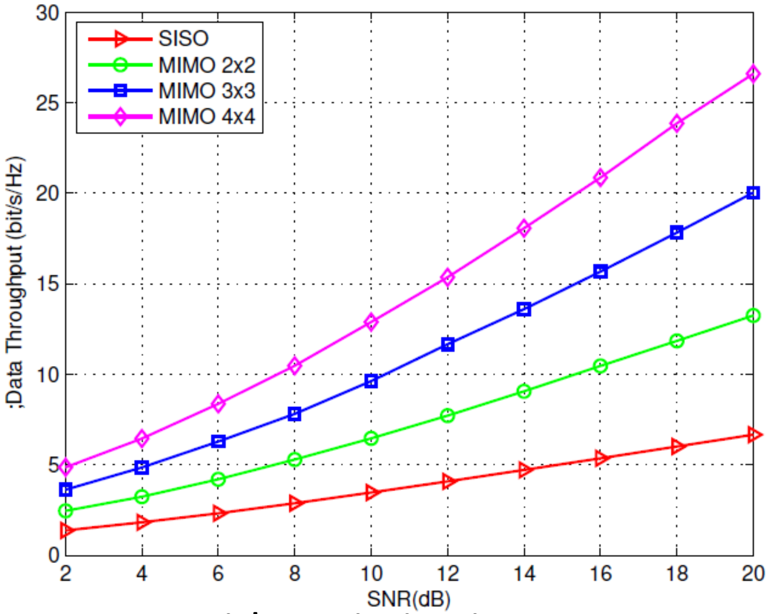
BIBLIOGRAPHY

Leal et al. "Genetic Algorithm Optimization Applied to the Project of MIMO Systems"

- MIMO: Multiple Input Multiple Output, i.e., multiple antennas at transmitter and receiver
- Exploits multipath propagation
- Allows to have higher throughput and coverage area
- GA is used for optimization
- Goal: maximize throughput by varying the distance among antennas



Without optimization



With optimization

Politecnico di Torino
1859

Thank you for your attention!

DET

Politecnico di Torino

1859

GENETIC ALGORITHM

ANALYSIS AND EFFECTIVENESS EVALUATION IN SOLVING REAL-WORLD PROBLEMS

Department of Electronics and Telecommunications
01RGBRV Optimization methods for engineering problems
Prof. M. Repetto

Antonio Calagna, Giuseppe Di Giacomo, Francesco Gabriele and Deborah Volpe

June 7, 2022