KNAPSACK PROBLEM 01

- Fixed capacity backpack
- Objects list, containing weight and value
  - What set of objects maximizes the backpack value, with the maximum capacity constraint?

OBJECTIVES

- Confront different optimization approaches
  1. Analytic "*Greedy" approach*
  2. Genetic approach:
     A. Classical
     B. Quantum

```
oggetti = pd.DataFrame.from_records([
    ['compass', 70, 135],
    ['water', 73, 139],
    ['sandwich', 77, 149],
    ['glucose', 80, 150],
    ['tin', 82, 156],
    ['banana', 87, 163],
    ['apple', 90, 173],
    ['cheese', 94, 184],
    ['beer', 98, 192],
    ['suntan cream', 106, 201],
    ['camera', 110, 210],
    ['T-shirt', 113, 214],
    ['trousers', 115, 221],
    ['umbrella', 118, 229],
    ['note-case', 120, 240]])
```

DATASET

- 15 object
- Best solution:
  - [1,0,1,0,1,0,1,1,1,0,0,0,0,1,1]
  - Value 1458 e Weight 749.

ANALYTIC APPROACH

- Define a metric $Q = \dfrac{\text{Value}}{\text{Weight}}$
- Objects are ordered depending on Q
- Progressively fill the bag with objects, starting from best to worse Q

ANALYTIC APPROACH (2)

- Best solution
  - [1,1,1,1,1,1,1,0,0,1,0,0,0,0,0]
  - Value 1441, Weight 740.

- NOTE: According to the initial ordering it would be [1,1,1,0,0,0,1,1,1,0,0,0,0,1,1].

Nome	Peso	Valore	Quality
note-case	120	240	2.000000
beer	98	192	1.959184
cheese	94	184	1.957447
umbrella	118	229	1.940678
sandwich	77	149	1.935065
compass	70	135	1.928571
apple	90	173	1.922222
trousers	115	221	1.921739
camera	110	210	1.909091
water	73	139	1.904110
tin	82	156	1.902439
suntan cream	106	201	1.896226
T-shirt	113	214	1.893805
glucose	80	150	1.875000
banana	87	163	1.873563

Q-based ordering

GENETIC ALGORITHM

- Based on natural evolution
- Imitation of the natural processes of:
  - Selection
  - Crossing over
  - Mutation
- Define and minimize a *fitness function*

TWO CROSSOVER APPROACHES

Two-Parent

- Select whole population
- Assign the role of parent 1 and 2 to individuals with an even/odd index
  - **Crossover Probability (PC)** to establish which couple can generate offspring

Multi-parent

- Only a subset of the population is chosen as parent
  - **Crossover Probability (PC)** to establish which individual can be part of the subset

IMPLEMENTED CROSSOVER OPERATORS

Two-Parent

- One-point crossover
- Two-point crossover
- Uniform crossover

Multi-parent

- Uniform Multi-parent crossover
- Quantum Mating Operator

ONE-POINT CROSSOVER

- Random choice of a point on the parent sequence
- Exchange portion of the parents' genome to create offspring

0	1	1	0	0	0	0	1	0	1
1	0	0	1	0	1	1	0	0	0
0	1	1	0	0	1	1	0	0	0
1	0	0	1	0	0	0	1	0	1

TWO-POINT CROSSOVER

0	1	1	0	0	0	0	1	0	1
1	0	0	1	0	1	1	0	0	0
0	1	1	0	0	1	1	1	0	1
1	0	0	1	0	0	0	0	0	0

- Random choice of two points on the parent sequence
- Exchange portion of the parents' genome to create offspring

UNIFORM CROSSOVER

- Presence of a mask (green line)
  - Establishes from which parent the gene is taken (0 from parent 1, 1 from parent 2)

0	1	1	0	0	0	0	1	0	1
1	0	0	1	0	1	1	0	0	0
1	0	0	1	0	0	1	1	1	0
1	1	1	1	0	0	1	0	0	1
0	0	0	0	0	1	0	1	0	0

MULTI-PARENT ALGORITHM

- Genetic algorithms with crossover operators which take a gene from more than one parent to produce offspring
- Select portion of parents genome to produce offspring, in analogy with two-parent mechanisms

UNIFORM MULTI-PARENT CROSSOVER

0	1	1	0	0	0	0	1	0	1
1	0	0	1	0	1	1	0	0	0
0	0	1	0	1	0	1	1	0	0
2	1	1	2	0	2	1	2	0	1
0	0	0	0	0	0	1	1	0	0

- Mask (green line) establishes from which parent the gene is taken (0 from parent 1, 1 from parent 2, 2 from parent 3)

MUTATION

- Alteration of some gene, regardless of the parents genome
- Useful to improve genome's variability
  - Improves fitness value
  - Enlarges the search space

MUTATION (1)

PME (Probabilità Mutazione Esterna):

- Probability that a single individual manifests the mutation

PMI (Probabilità Mutazione Interna):

- Probability for a single gene to be mutated.

PMQ (Probabilità Mutazione Quantistica):

- Probability for a single gene to be mutated. It also defines the rotation angle (in units of π)

BIT FLIP MUTATION

- We establish if an individual will have mutations (PME)
  - **If it does:** generate a random number $x \in [0,1]$
  - If $x <$ PMI, switch the gene

0	1	1	0	0	0	0	1	0	1
0.9	0.7	0.8	0.4	0.5	0.3	0.6	0.2	0.5	0.4
0	1	1	1	0	1	0	0	0	1

Ex. With PMI=0.45

QUANTUM MATING OPERATOR

Operator to be used in a hybrid way

- Runs on quantum circuit
- Results used in a classic genetic algorithm

Architecture:

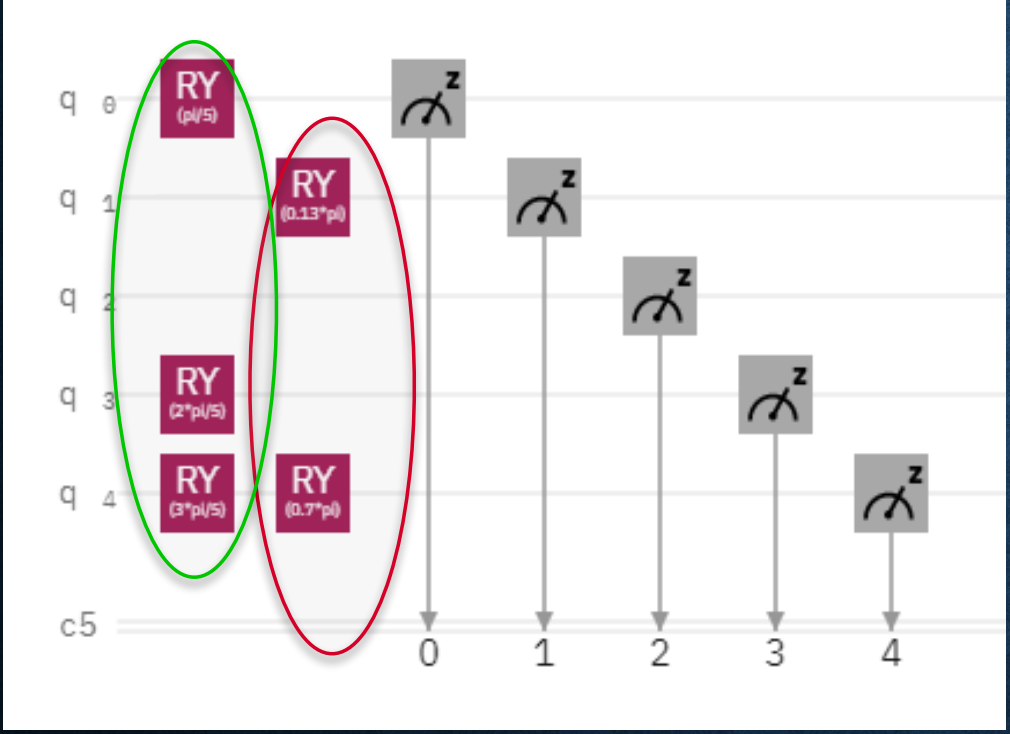
- Multi-parent Crossover Operator
- Mutation Operator

Based on Y rotation gate

QMO

0	0	0	0	1
1	0	0	0	1
0	0	0	1	0
0	0	0	1	1
0	0	0	0	0
0.2	0	0	0.4	0.6
0	0.13	0	0	0.7

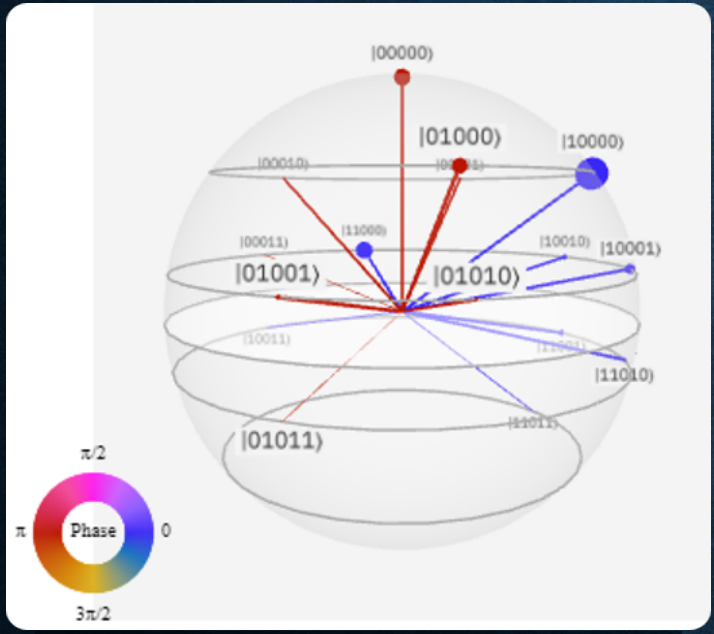
- Select a parent subset based on PC
- Mating Mask
  - Formed from the frequencies f_i of 1 in parents genes
  - Provides rotation angles along Y on Bloch sphere $\theta_i = f_i \pi$
- Mutation Mask
  - Formed with random numbers
  - Multiples $[0,1]$ of π



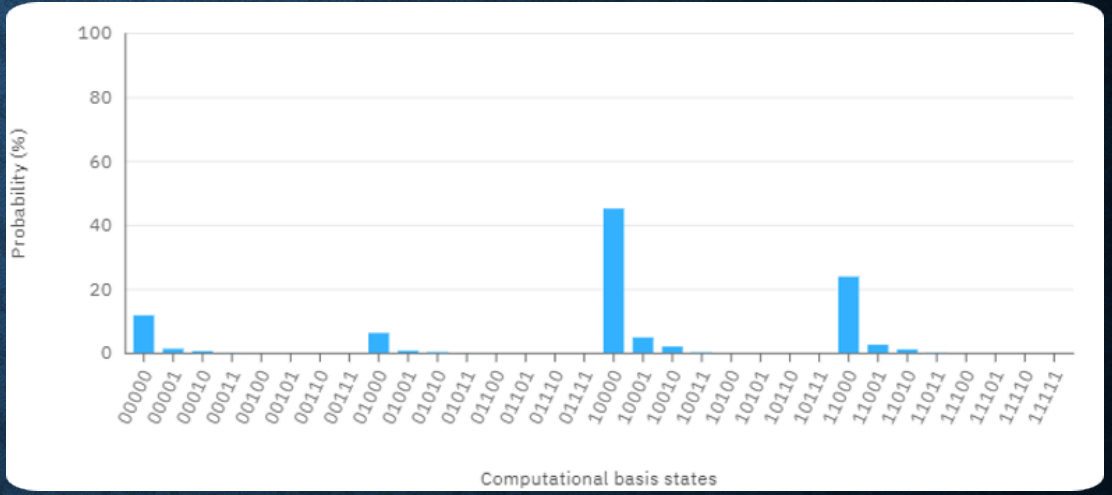
Quantum Circuit

- Input qubit in $|0\rangle$
- First, rotation due to mating processes
- Second, rotations due to mutation
- C5 is the classical register

BEFORE THE MEASUREMENT

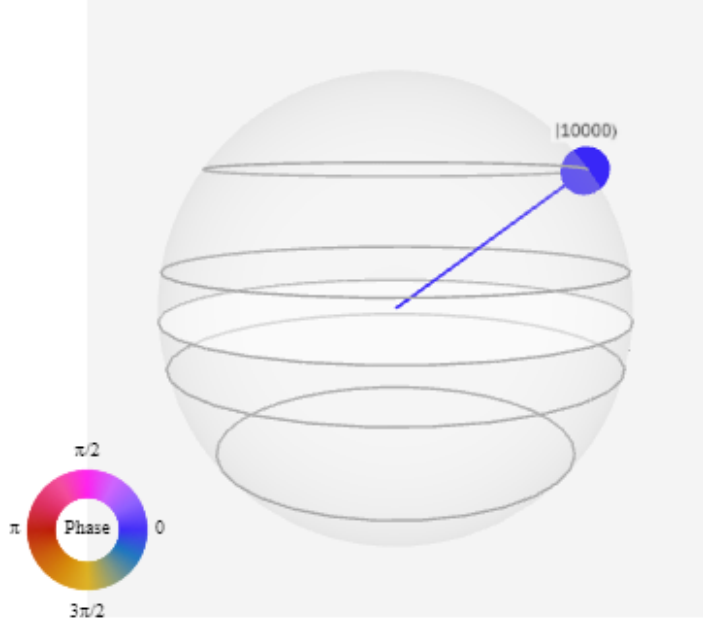


System state on the Bloch sphere

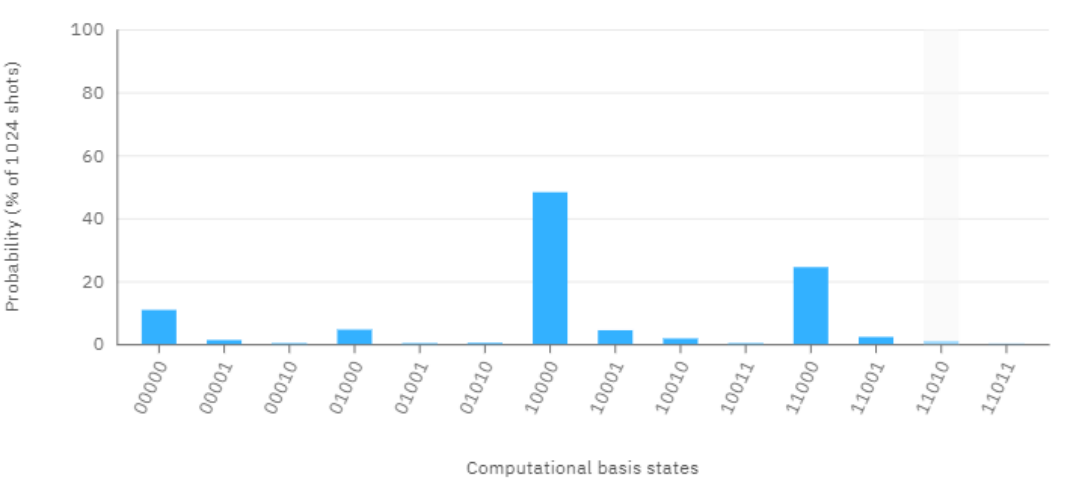


Probability distribution for the states of the system

AFTER THE MEASUREMENT



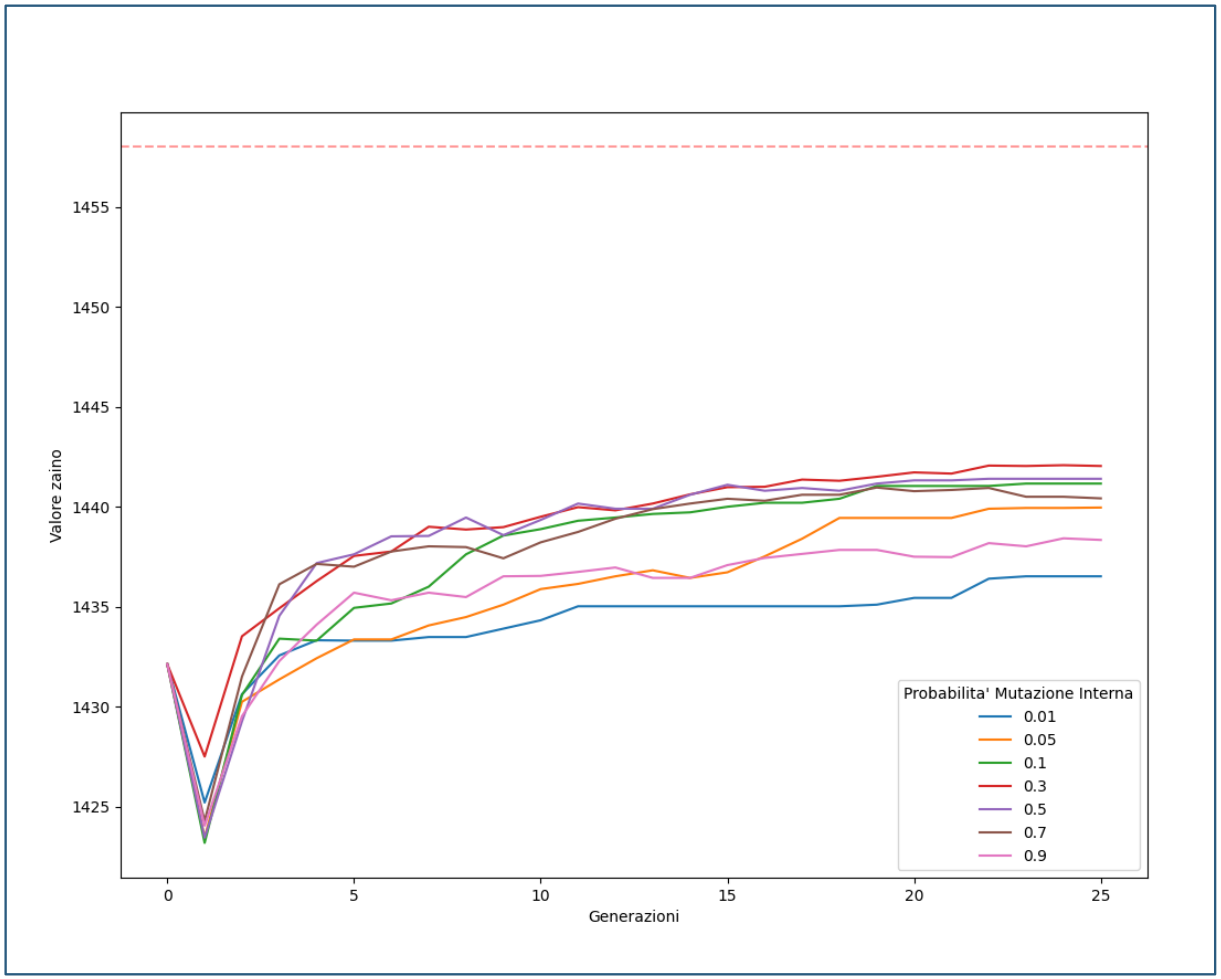
System state on the Bloch sphere



Probability distribution for the states of the system

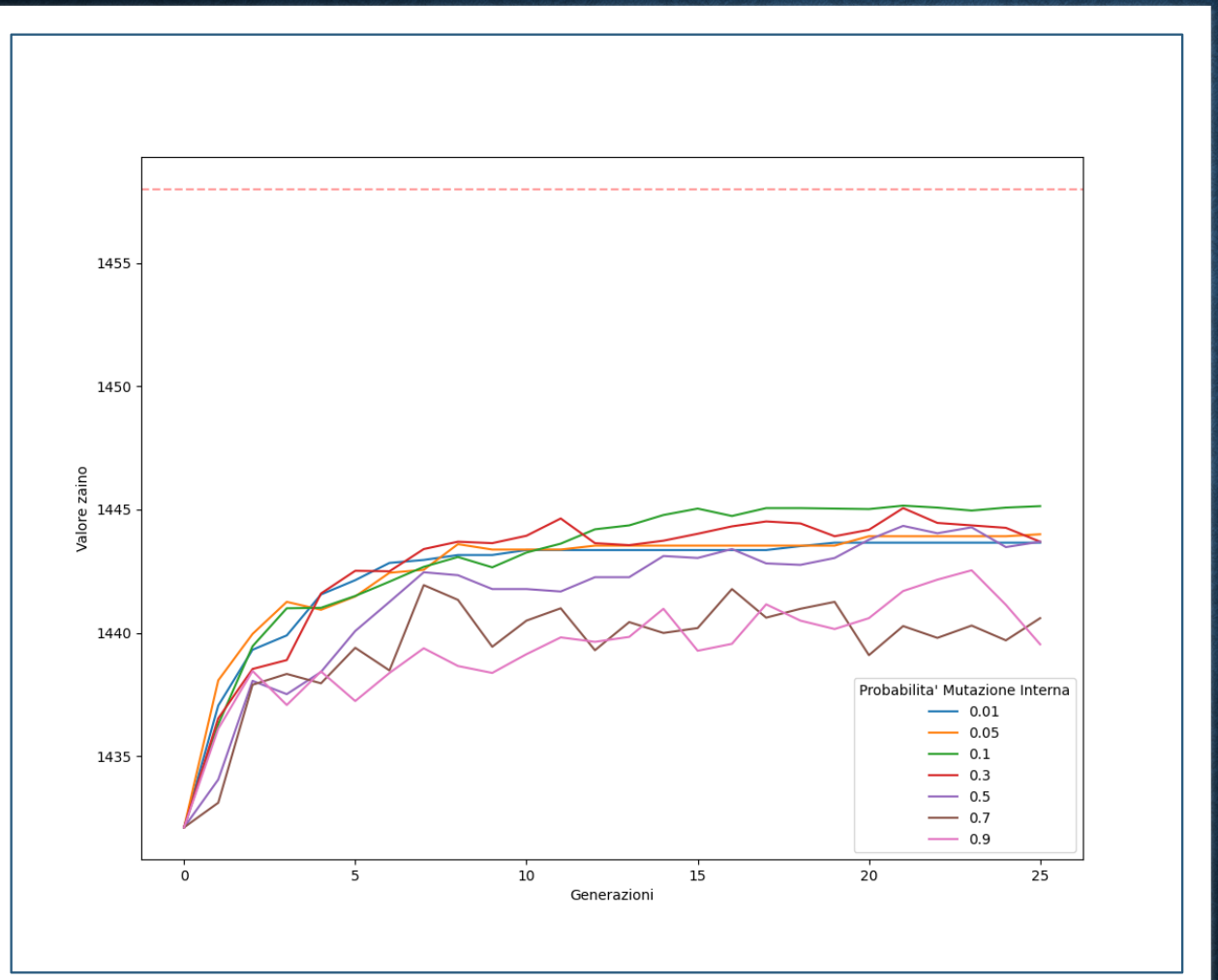
PARAMETER TUNING

- **Problem**:
  - How to get the best parameters?
- Sequential test of different combinations
  - Select the set of parameters which enables to get the highest value for the backpack
- Parameters to tune:
  - PC
  - PME
  - PMI
  - PMQ
  - (Only for UC) Uniform Probability
- Constant parameters:
  - Population → 30
  - Generations → 25
  - Number of seeds→ 30



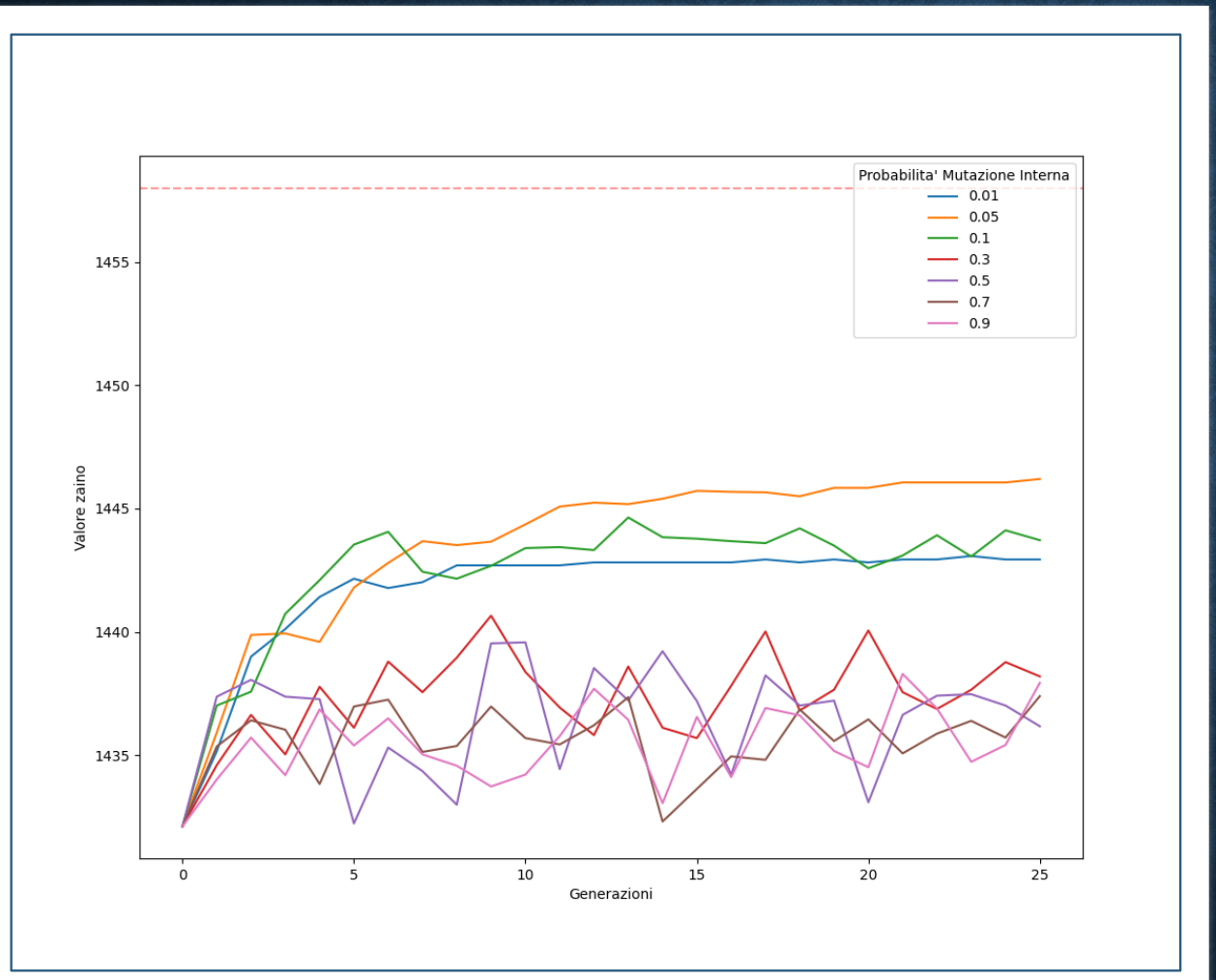
TUNING ONE POINT CROSSOVER

- Starting with OPC, fix the value for PC and PME and varied PMI:
  - PC = 0,1
  - PME = 0,1



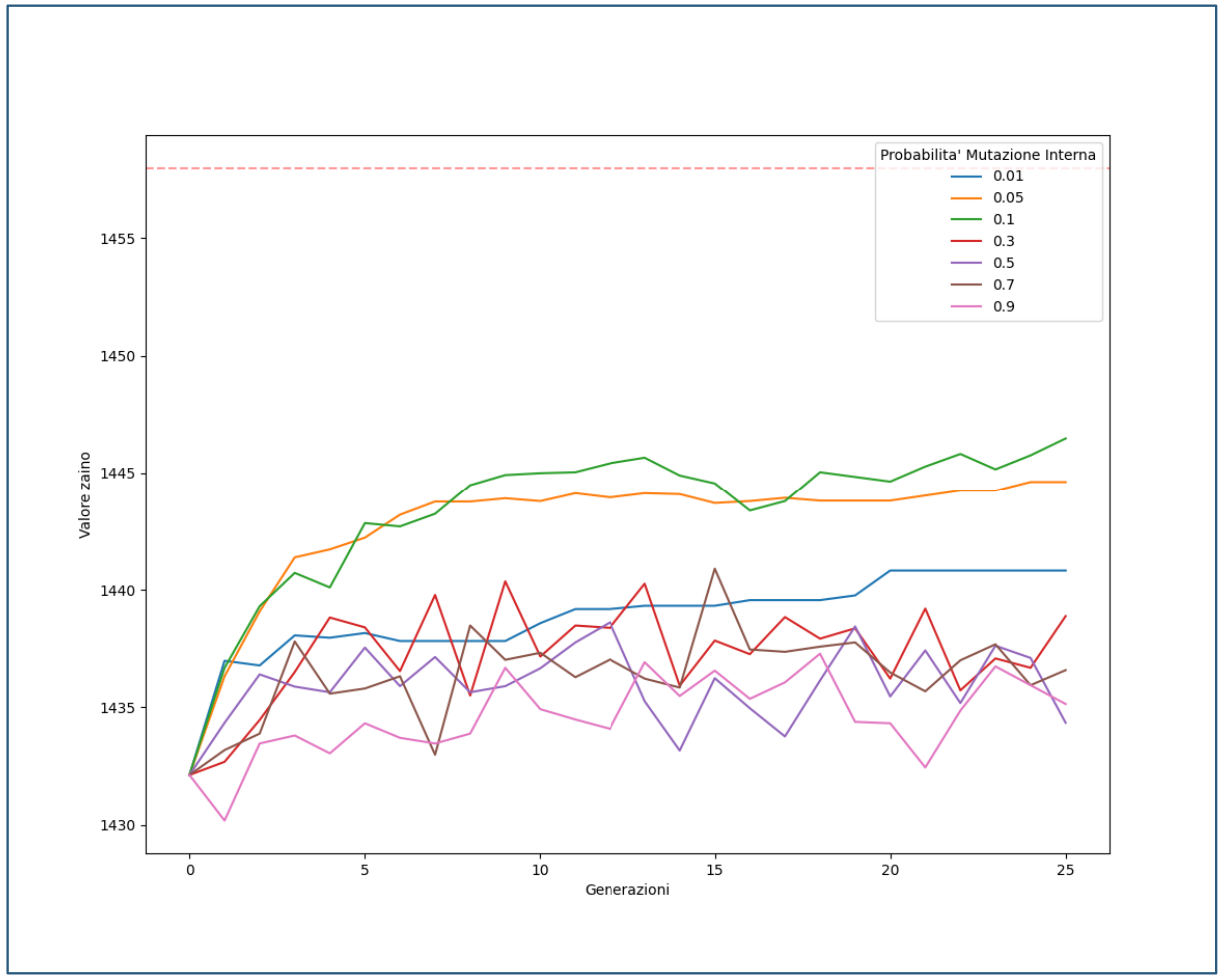
TUNING ONE POINT CROSSOVER

- PC = 0,5
- PME = 0,5



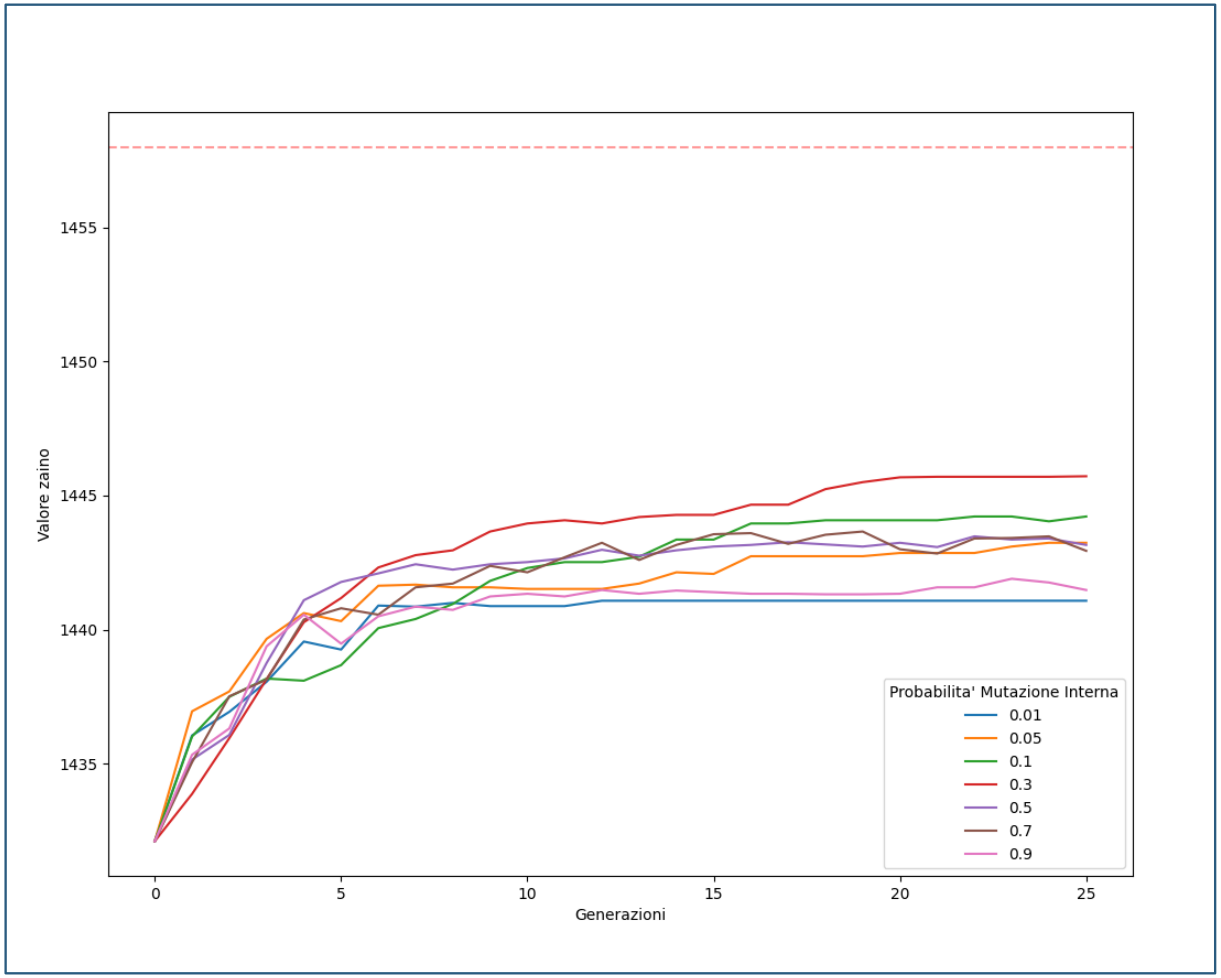
TUNING ONE POINT CROSSOVER

- PC = 0,8
- PME = 0,9



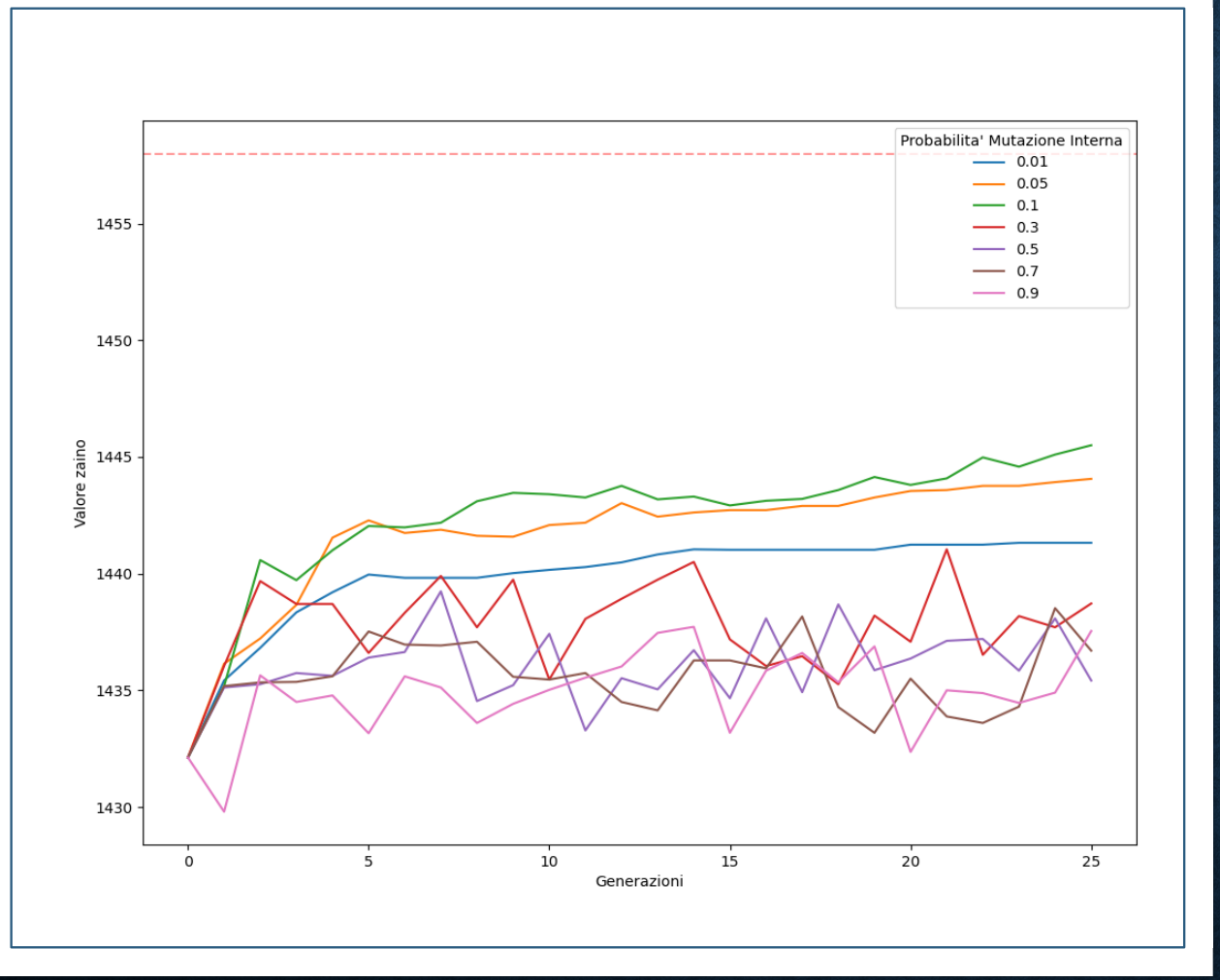
TUNING ONE POINT CROSSOVER

- PC = 0,1
- PME = 0,9



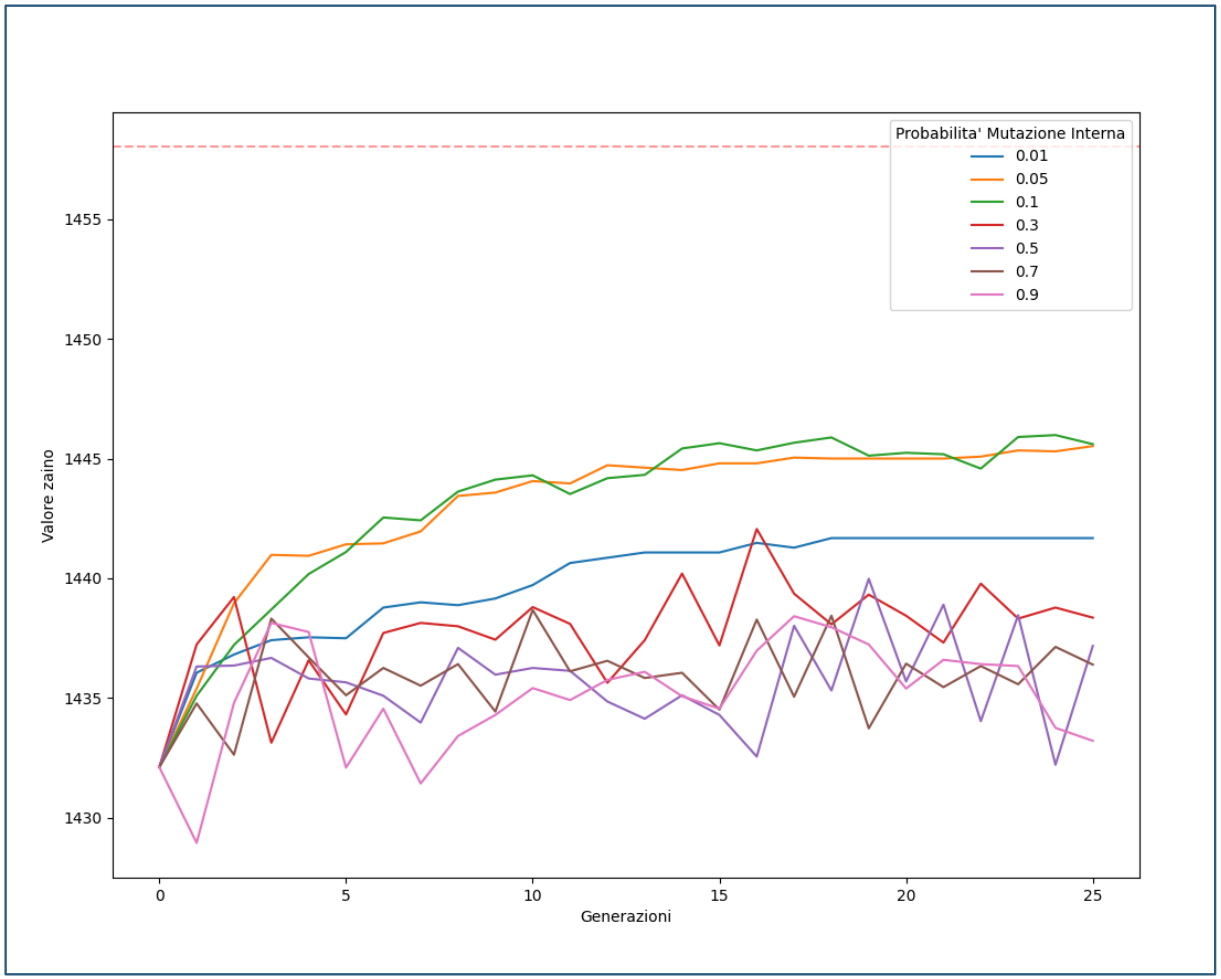
TUNING ONE POINT CROSSOVER

- PC = 0,8
- PME = 0,1



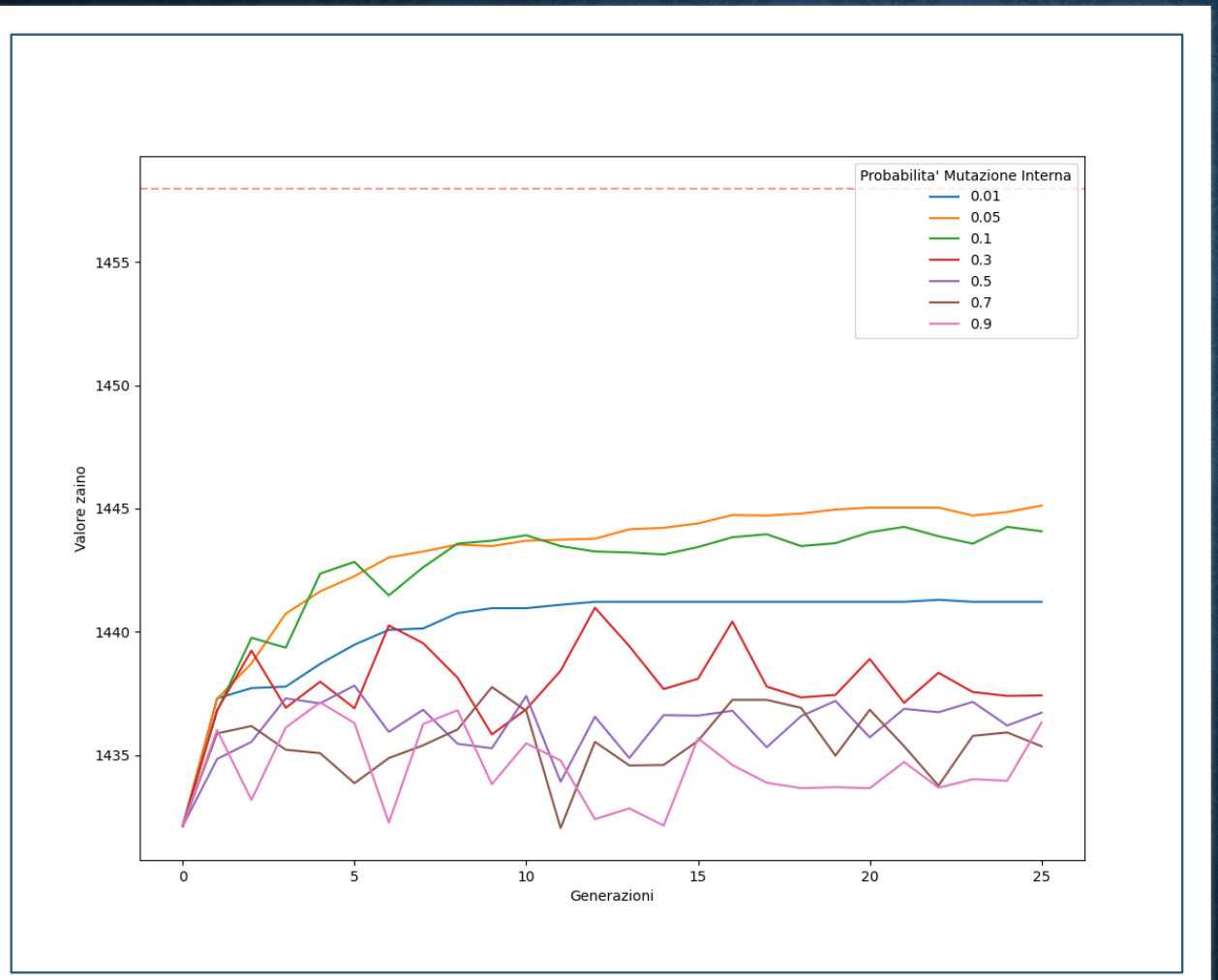
TUNING TWO POINT CROSSOVER

- PC = 0,1
- PME = 0,9



TUNING UNIFORM CROSSOVER

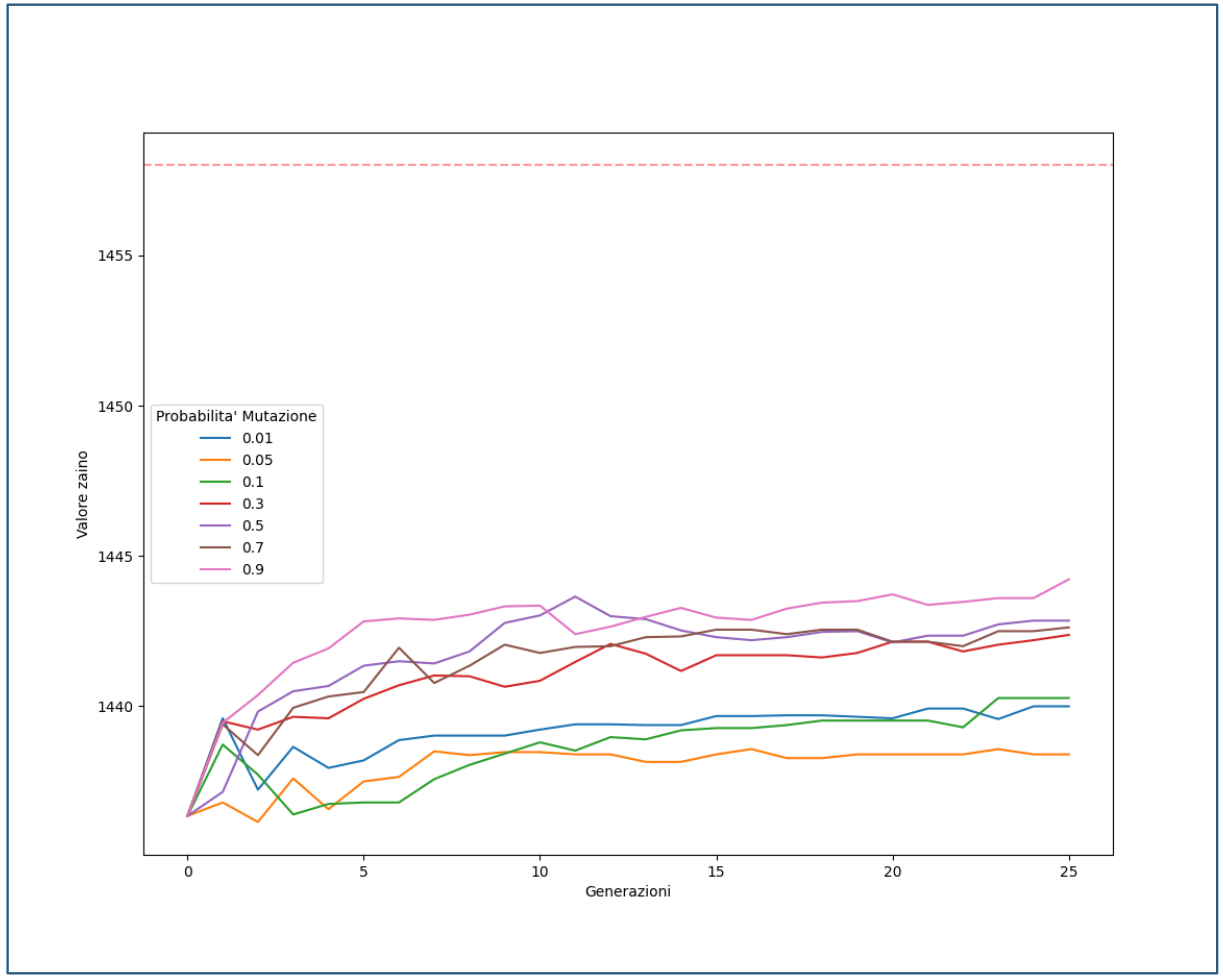
- PC = 0,7
- PME = 0,3
- PU = 0,5



TUNING MULTIPARENT CROSSOVER

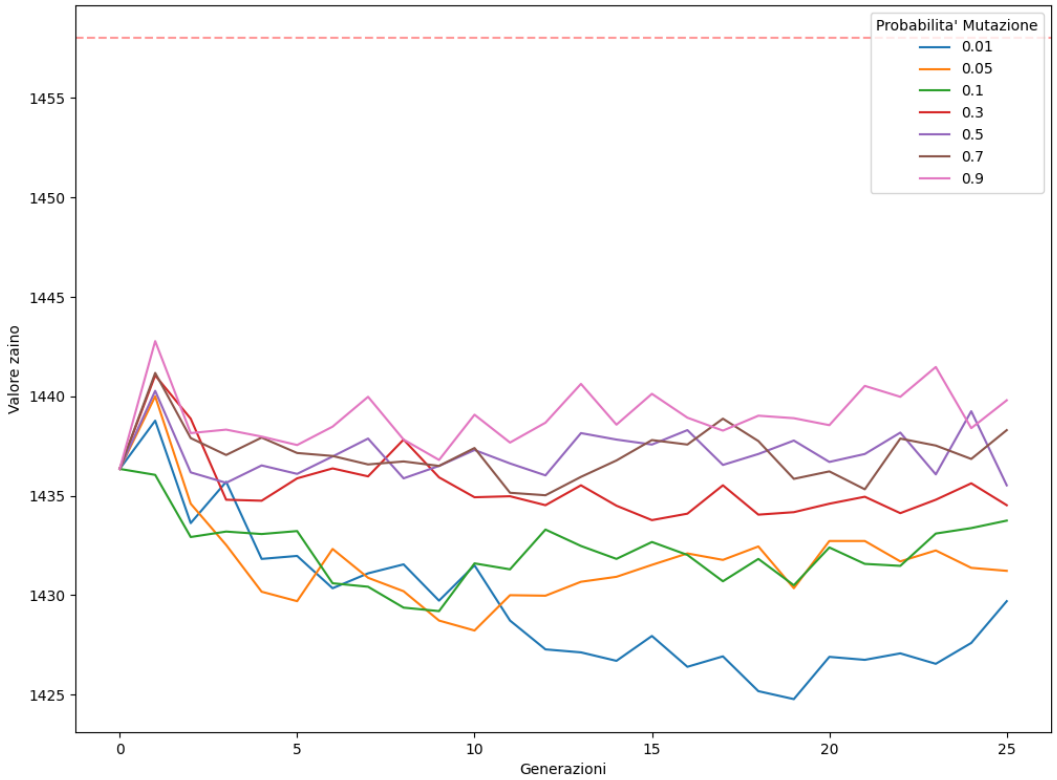
- PC = 0,1
- PME = 0,9

TUNING QMO



- PC = 0,3

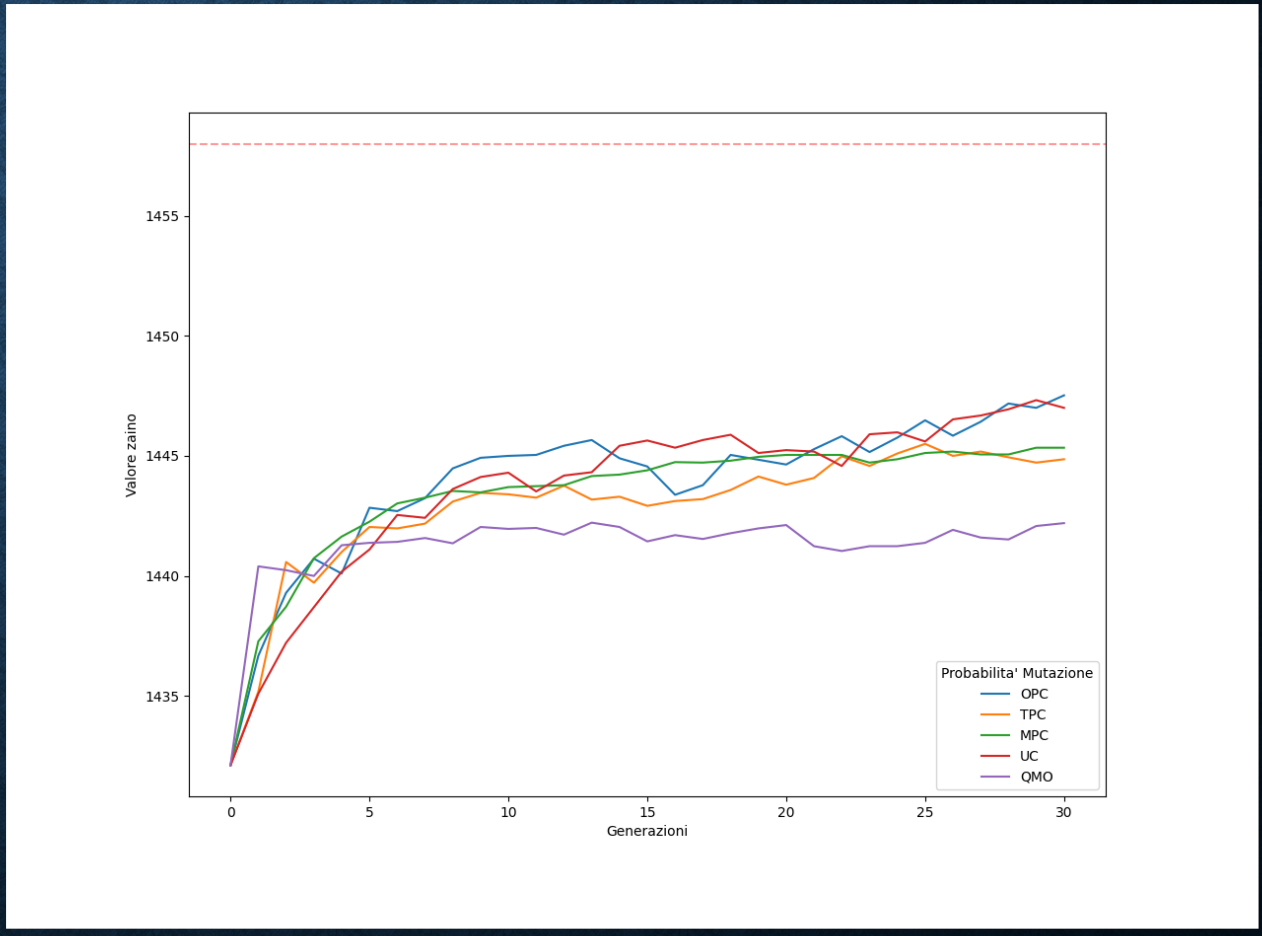
TUNING QMO



- PC = 0,8

COMPARISON BETWEEN ALGORITHMS

- Population 30
- 50 seeds
- 30 generations
- Runtime
  - Genetic algorithms → 30 seconds
  - QASM simulation → 5 minutes

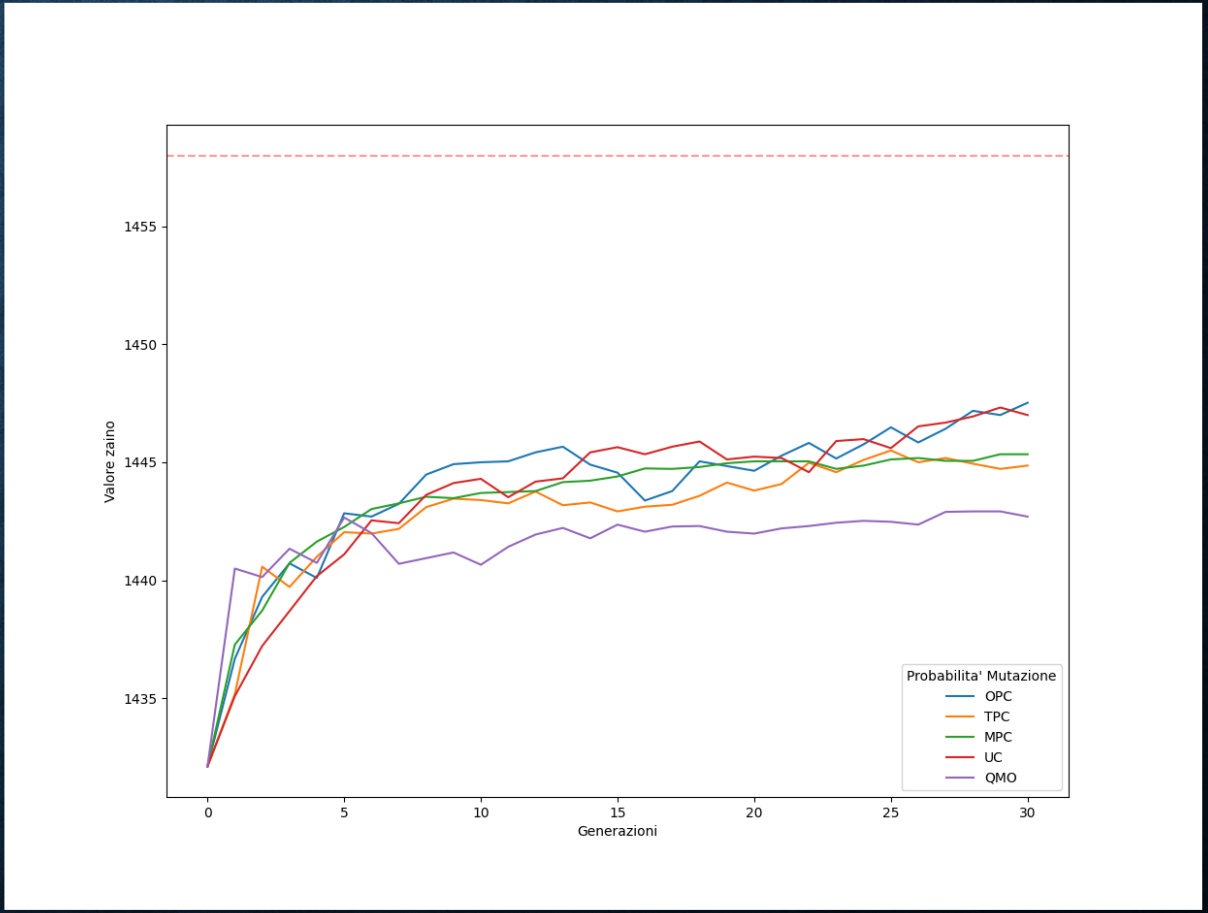


QUANTUM NOISE

- Decoherence problem
- System is not isolated
  - Interaction with ambient
- QC suffer a lot from noise

COMPARISON BETWEEN ALGORITHMS

- Population 30
- 50 seeds
- 30 generation
- Runtime
  - Genetic algorithms → 30 seconds
  - FAKEGUADALUPE simulator → 5.5H



CONCLUSIONS

Quantum Algorithm	Performance	Comparison with analytical approach
• No computational advantage • Runtime performance is worse • Suffers from low values of PMQ e high ones of PC	• Comparable	• Every algorithm is better

BACKUP