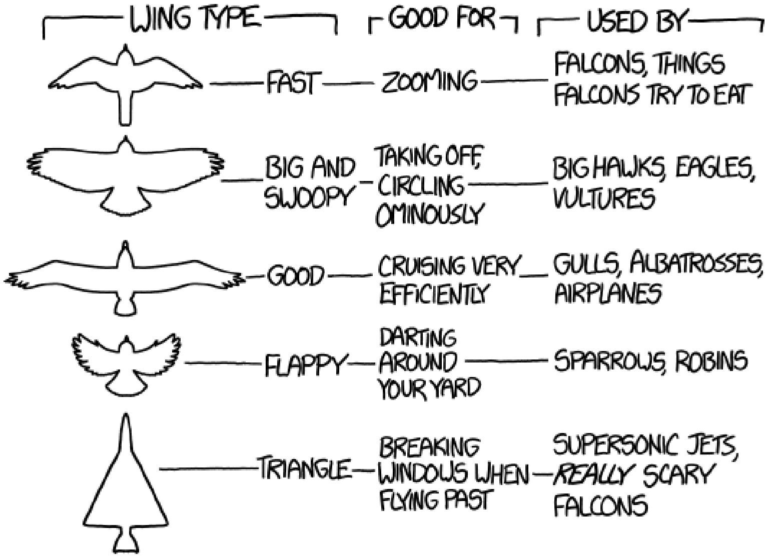
ECTA Homework 3
Shape Matching Problem

Erick Kramer, Mihir Patil
erick.romero@smail.inf.h-brs.de , mihir.patil@smail.inf.h-brs.de

June 13, 2018


WING TYPE
GOOD FOR
USED BY
FAST
ZOOMING
FALCONS, THINGS FALCONS TRY TO EAT
BIG AND SWOOPY
TAKING OFF, CIRCLING OMINOUSLY
BIG HAWKS, EAGLES, VULTURES
GOOD
CRUISING VERY EFFICIENTLY
GULLS, ALBATROSSES, AIRPLANES
FLAPPY
DARTING AROUND YOUR YARD
SPARROWS, ROBINS
TRIANGLE
BREAKING WINDOWS WHEN FLYING PAST
SUPERSONIC JETS, REALLY SCARY FALCONS

To optimize wings, it is useful to start with known designs and then adjust them to specific tasks. How to convert from one representation to another? In this assignment you will convert the 4-dimensional NACA wing specific representation into a general B-Spline representation with 32 control points. You will compare 2 evolutionary methods, Genetic Algorithms and Evolution Strategies.

1 Assignment Description

Shape Matching Problem

1. Write your own Genetic Algorithm to solve the shape matching problem
2. Write your own version of the Covariance Matrix Adaptation Evolution Strategy (CMA-ES) to solve the shape problem.
3. Compare the performance of the two algorithms on three airfoils (NACA airfoil shapes: 0012, 5522, 9735). Is there a significant difference between a GA and an ES?

- Grading Scheme
  - □ Genetic Algorithm (20 pts)
    - □ Bitstring *or* Real-Valued (20 pts)
    - □ Bitstring *and* Real-Valued (+10pts)
  - □ Evolution Strategies (60 pts)
    - □ ES with 1/5 rule (30 pts)
    - □ CMA-ES with out evolution paths (30 pts)
    - □ CMA-ES with evolution paths (+10)
  - □ Comparisons (20 pts)
    - □ Big beautiful wall of data

2 Submission Instructions

To be perfectly clear we expect two submissions to LEA:

1. 1 PDF (report) – a modified version of your submission PDF, with your own code snippets, figures, and responses inserted
2. 1 ZIP (code and data) – a .zip file containing all code use to run experiments (.m files) *and* resulting data as a .mat file
3. Make sure to follow the naming scheme HW_NUMBER_LASTNAME1_LASTNAME2.suffix
4. → A valid name would be HW_02_Smith_Fernandez.pdf
5. Make sure both team members use the same filename!

3 The Assignment

3.1 Genetic Algorithm

- Genetic Algorithms are typically represented by a string, this string could take many forms, such as bitsrings or real-valued numbers. What are the advantages and disadvantages of each encoding in this application? How would you convert each into 32 real-valued numbers? How could you perform crossover and mutation in each? Which do you think would be best?

  1. Bitstring
     - Advantages:
     + Provide computational improvements, as mutation by randomly flipping bits is faster than generating numbers.
     - Disadvantages:
     + Coding and decoding functions might introduce additional multimodality (Bck et al.)
     + No control over the value getting by flipping random bits for a floating point number. - Mutation:
     + Flip random bits based on the mutation probability. The problem is that we do not really now if by flipping that bit we will get a value in between the desired range.
     - Crossover:
     + One point crossover by splitting the individual on a random point, making sure that the split point does not land in between a gene representation, and getting the remaining genes for a second individual.
     - A function is used to convert the individual into a bit representation and another one gets it back to be able to calculate the corresponding fitness.
     - As the application requires floating points, and negative values are allowed, gray coding was not possible. An alternative was found on [1].
     - We believe this will get better results, because of the computational complexity.

  2. Real-valued
     - Advantages:
     + More readable.
     + There is no need to encode and decode the population to calculate the fitness value.
     -Disadvantages:

[1] https://www.mathworks.com/matlabcentral/answers/25549-convert-floating-point-to-binary

+ It has more computational cost.
- Mutation:
+ Randomly changing the gene to a value in the range of ±0.5. - Crossover:
+ One point crossover by splitting the individual on a random point and getting the remaining genes for a second individual.

- Implement the encoding you think will perform best and plots its median performance over 20 iterations, using 20,000 evaluations.
  Your plot here

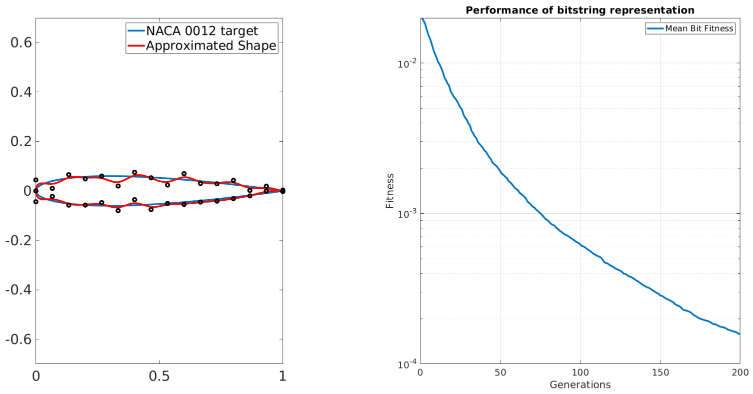
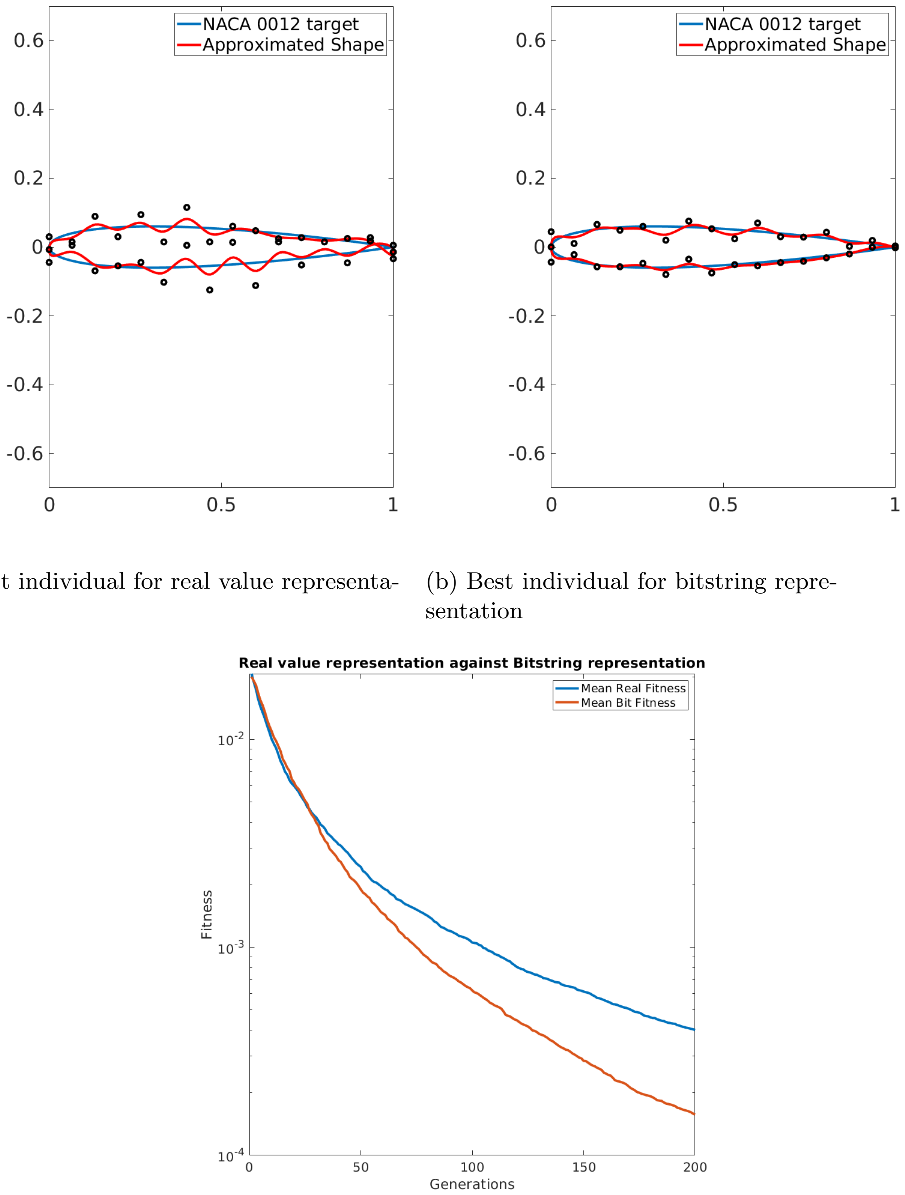


Figure 1: a) Performance of the GA having a bit string representation. b) Best individual over 20 runs.

- ****Extra Credit**** Implement both encodings and compare them on matching task for one of the shapes. Is one significantly better? Can you explain why?
  - Run-time:
  + Real value representation: 330 seconds
  + Bitstring representation: 335 seconds.
  Having a higher run-time for the bitstring was expected due to the many conversions that are required for the individuals in order to compute the crossover, mutation and fitness value calculation.

  As it can be appreciated from the resulting images, the bitstring representation obtained a much better result in comparison with the real-value representation. We looked for a concrete reason behind the better performance of the bitstring, but we only found that authors tend to refer to the bitstring as the optimal representation. But, we could not find an idea that could back up this claim.

Perhaps, the way mutation occurs differently than the real value representation could be the key aspect of the better performance of the bitstring representation.



(a) Best individual for real value representation.

(b) Best individual for bitstring representation

(c) Comparison of the mean fitness for real value representation and bitstring..

Figure 2: GA implementation over 20,000 evaluations

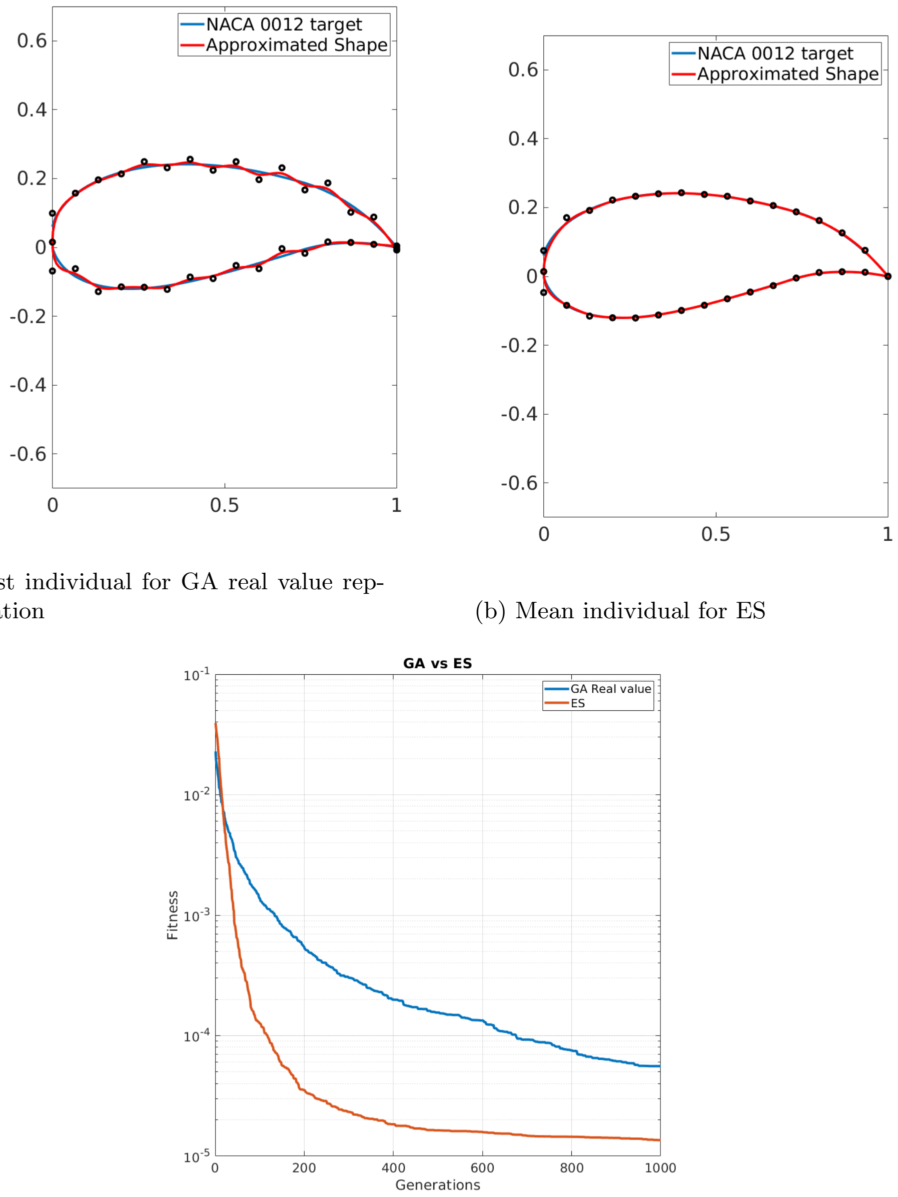
3.2 Evolution Strategies

- CMA-ES is an advanced version of ES. As a first step implement a simple ES first.
    - In this ES mutation of all parameters should have equal strength which is adjusted by the 1/5th rule.
      (*every N generations change mutation strength: if* $> 1/5th$ *of mutations resulted in an better fitness (i.e. a best solution) increase mutation strength, otherwise decrease mutation strength*). Test your implementation on one of the shapes.
    - Compare to your best GA. Is one significantly better than the other?
      ES converged much faster than GA. ES required less generations to find a good solution.

- Now program CMA-ES without evolution paths.
    - Compare to your ES results. Is a there a significant improvement?
      To our surprise, there was no improvement in using CMA-ES in comparison to ES. Perhaps, due to the tweaking we performed in the ES to speed up the calculation.

- ****Extra Credit**** Now program CMA-ES with evolution paths.
    - Compare to your previous CMA-ES results. Is a there a significant improvement?

      Yes, the CMA-ES with evolution paths converge much faster than the one without evolution paths.

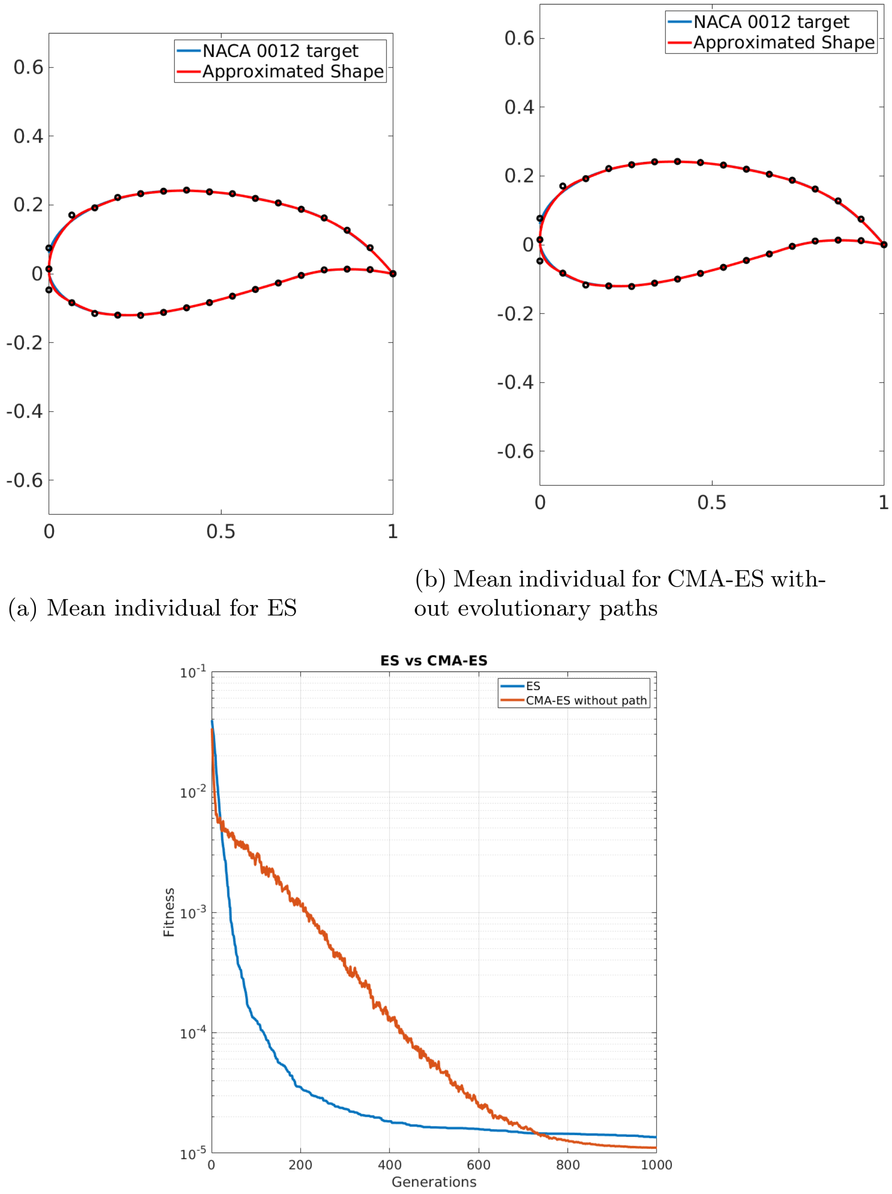


(a) Best individual for GA real value representation

(b) Mean individual for ES

(c) Performance plot.

Figure 3: GA vs ES on wing 3 for 20 runs

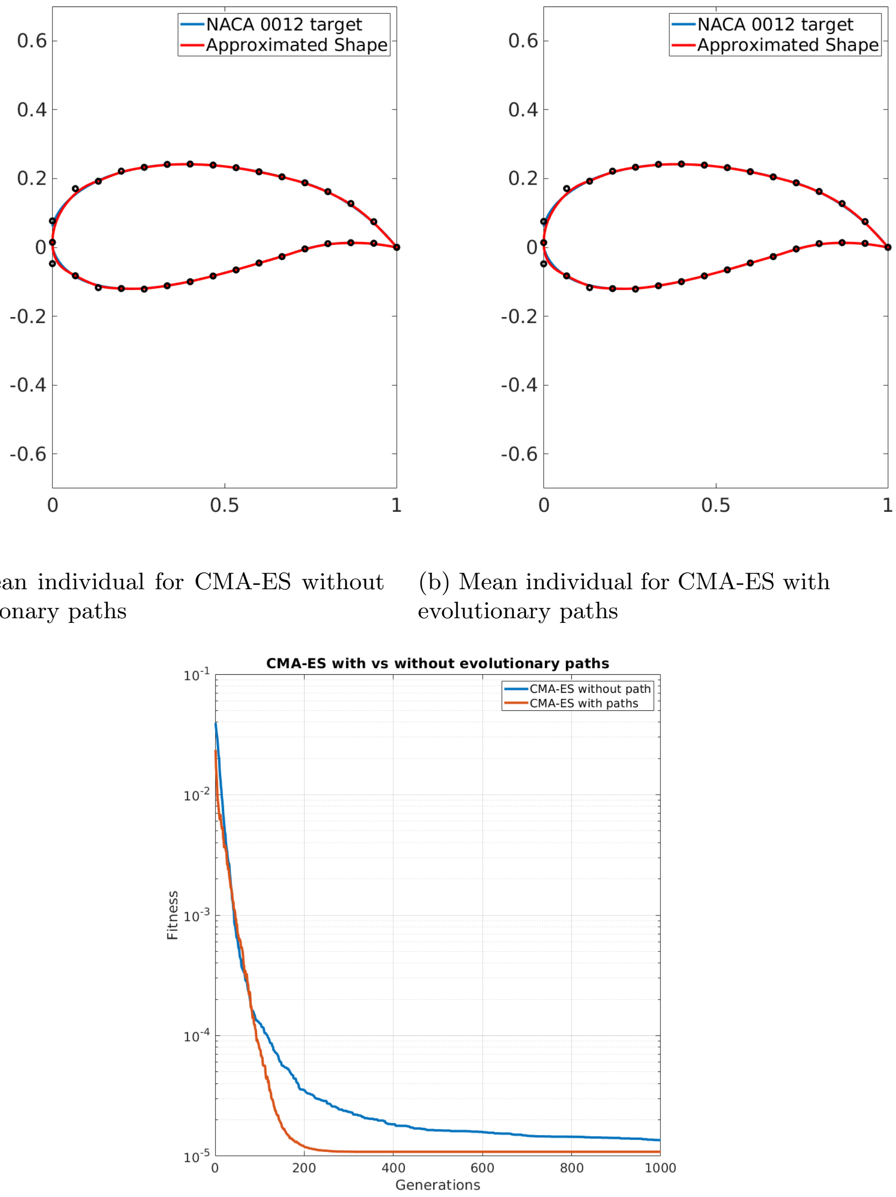


(a) Mean individual for ES

(b) Mean individual for CMA-ES without evolutionary paths

(c) Performance plot.

Figure 4: ES vs CMA-ES without evolutionary paths on wing 3 for 20 runs



(a) Mean individual for CMA-ES without evolutionary paths

(b) Mean individual for CMA-ES with evolutionary paths

(c) Performance plot.

Figure 5: CMA-ES with vs without evolutionary paths on wing 3 for 20 runs

3.3 Comparisons

Produce one plot which shows the performance of each algorithm. Run each algorthm 20 times on each shape (NACA airfoil shapes: 0012, 5522, 9735), with a budget of 20,000 function evaluations. For each run record the best ever found individual at each evaluation. For each algorithm plot the median fitness of this best ever individual. This may take some time, write a script to run and collect this data. The following should be on this plot (leaving out any algorithms you chose not to implement):

1. Real-Valued GA
2. ES
3. CMA-ES (without evolution paths)
4. CMA-ES (with evolution paths)

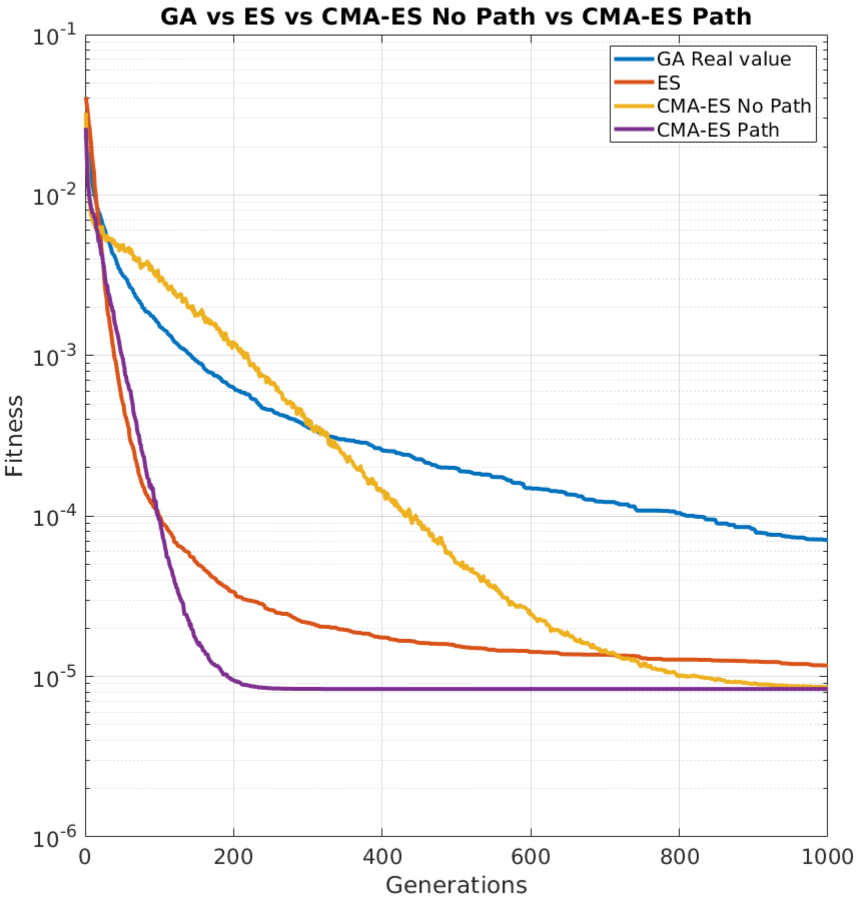


Figure 6: Performance comparison between:
a) GA with real value representation
b) ES with 1/5 rule
c) CMA-ES without evolution paths
d) CMA-ES with evolution paths.