Computer Work

Evolutionary Multi-objective Optimization

NSGAII

ULPGC

Master's Degree in Intelligent Systems and Numerical Applications in Engineering

Optimal Engineering Design

Leopoldo Lopez Reveron

deliver date: 23/12/2021

Objectives:

1- To use an evolutionary multi-objective optimization software.

2- To execute different parameters in a test case: variation of population size and mutation rate should be considered at minimum.

3- To analyze and evaluate the effect of the chosen parameter set in the algorithm convergence by using the hypervolume metric.

4- To find a combination of parameter for each problem, which achieves good results (at least population size and mutation rate should be tested in a 3x3 combination set: 3 values of population size and 3 values of mutation rate, resulting in 9 parameter combinations).

Procedure:

1- In the computer laboratory, students follow the professor instructions.

2- They allow to guide step by step in a common mathematical test case.

3- Later, the students will apply these to other functions chosen by themself.

4- A report must be written with the process, results, and conclusions.

The minimum content of the report:

1- includes problem definition and name

2- mathematical equation of the assigned problem.

3- For each combination of parameters tested, a figure with the non-dominated final front and convergence curve of hypervolume indicator.

4- Figures where the comparison of hypervolume convergence are detailed, as well as explanation and justification of the best set of parameters tested under the perspective of balance between exploration and exploitation.

5- Same number of total numbers of fitness function evaluations should be tested in each execution (maximum value of 25.000 evaluations).

***Use random seed = 0.52**

Canadian Traveller Problem

Definition:

The Canadian Traveller Problem (CTP) is to find the shortest route from a source to a destination under uncertain conditions.

C.-S. Liao, Y. Huang / Theoretical Computer Science 530 (2014) 80–88 81

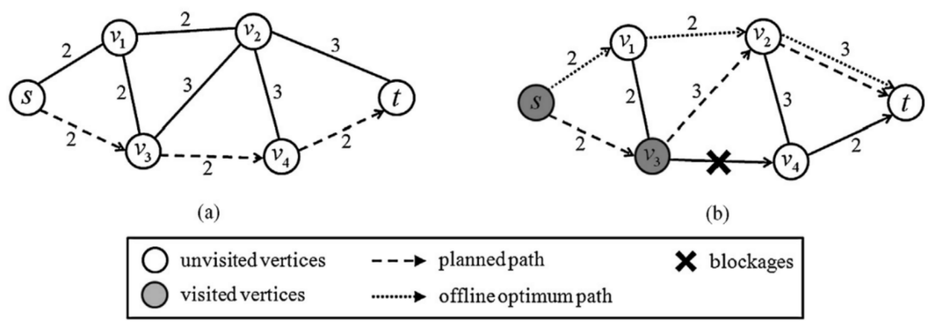


Fig. 1. An example that illustrates an unpredictable blockage occurring in a planned route.

Canadian Traveller Problem example

Mathematical definition:

CTP = (V, E, F, s, t, r).

exists a policy π such that for every realization (V, B) ∈ G(V, E, F),

the cost c(π, B) of the policy is no more than r times the off-line optimal, d_B(s, t).

Complexity:

the complexity of the problem and reported it to be PSPACE-complete

Pseudocode of CPT-8:

```
Procedure cycle_routing

    Alpha = 1e-32;       -> Division by 0 correction

    exp1 = (objetive[1] - e) * cos(theta) - objetive[0] * sin(theta);

    exp2 = (objetive[1] - e) * sin(theta) + objetive[0] * cos(theta);
    exp2 = b * PI * (exp2 ** c);
    exp2 = | sin(exp2) |;
    exp2 = a * (exp2 ** d);

    return (exp1 / ((exp2 - 1.0) + Alpha));

End procedure


Procedure ctp8

    theta = 0.1 * PI;
    Alpha = 1e-32;       -> Division by 0 correction
    a = 40.0;
    b = 0.5;
    c = 1.0;
    d = 2.0;
    e = -2.0;
    g = 1.0 + xreal[1];

    objetive[0] = xreal[0];
    objetive[1] = g * (1.0 – square_root(objetive[0] / (g + Alpha)));


    constraint[0] = cycle_routing(a, b, c, d, e, g, theta, objetive);


    theta = -0.05 * PI;            -> Second part
    a = 40.0;
    b = 2.0;
    c = 1.0;
    d = 6.0;
    e = 0.0;

    constraint[1] = cycle_routing(a, b, c, d, e, g, theta, objetive);

end procedure
```

Process

First, an adaptation of the code was made for its compilation in windows:

1- the adaptation of the paths due to the difference of reference levels within the file system.

2- Adjust the differences between the compilation version due to the difference of GCC version, specifically the adaptation of LongDouble to Double, it has been considered that it is not necessary so much precision for the concrete problem to treat CTP-8, as it was determined in the previous analysis in the exploration of the typical values of the problem.

Secondly, the creation of the parameter files for the solver has been programmed:

1- Python was chosen as the language for the development of the research infrastructure.

2- The parameter files contain the following meanings:

popsize: population size (a multiple of 4) (20 - 8.000)
ngen: Number of generations (25 - 500)
nobj: Number of objectives 2
ncon: Number of constraints 2
nreal: Number of real variables 2
min_realvar[i]: minimum value of i^{th} real variable 0 -> 1

max_realvar[i]: maximum value of i^{th} real variable 0 -> 10
pcross_real: probability of crossover of real variable (0.6 - 1.0)
pmut_real: probability of mutation of real variable (0.4 - 0.6)
eta_c: distribution index for real variable SBX crossover (5 - 20)
eta_m: distribution variable polynomial mutation (5 - 50)
nbin: number of binary variables 0
choice: option to display the data realtime using gnuplot 0

3- The modified parameters and their ranges are:

Popsize: (20 - 8.000) step 320
ngen: (25 - 500) step 50
pcross_real: (0.6 - 1.0) step 10
pmut_real: (0.4 - 0.6) step 6
eta_c: (5 - 20) step 8
eta_m: (5 - 50) step 10

4- This choice has resulted in 8320 different configuration files.

Thirdly, due to the number of possibilities in the parameters, a wrapper has been made in python for the execution of the solver:

1- By means of the wrapper 3814 different configurations have been evaluated resulting in 89.5GB of information.

2- The information saved in disk of each evaluation consists of:

all_pop.out
best_pop.out
final_pop.out
hyperstats.out
hypervolume.out

3- This evaluation has taken 1.4 days of execution, however, a linear progression should not be made when determining how long each execution takes, because due to the order of parameter files, the files with higher population are at the end.

Fourth, faced with such a large amount of information, it was decided to use the hyper-stats metric to visualize the ten best and the ten worst combinations as follows:

1- Best evaluations:

Hyper-stat	Input File
(99986271.676388,	ctp8_2487)
(99986271.636553,	ctp8_4364)
(99986271.636425,	ctp8_4284)
(99986271.636366,	ctp8_4204)
(99986271.636211,	ctp8_4124)
(99986271.636077,	ctp8_4044)
(99986271.336364,	ctp8_2489)
(99986271.336344,	ctp8_2409)
(99986271.336331,	ctp8_2329)
(99986271.046653,	ctp8_4359)

Parameter set best evaluations:

ctp8_2487 -> [980 375 2 2 2 2 (0 1) (0 10) 0.6 0.4 13 25 0 0]
ctp8_4364 -> [1940 325 2 2 2 2 (0 1) (0 10) 0.8 0.4 5 45 0 0]
ctp8_4284 -> [1940 275 2 2 2 2 (0 1) (0 10) 0.8 0.4 5 45 0 0]
ctp8_4204 -> [1940 225 2 2 2 2 (0 1) (0 10) 0.8 0.4 5 45 0 0]
ctp8_4124 -> [1940 175 2 2 2 2 (0 1) (0 10) 0.8 0.4 5 45 0 0]
ctp8_4044 -> [1940 125 2 2 2 2 (0 1) (0 10) 0.8 0.4 5 45 0 0]
ctp8_2489 -> [980 375 2 2 2 2 (0 1) (0 10) 0.6 0.4 13 45 0 0]
ctp8_2409 -> [980 325 2 2 2 2 (0 1) (0 10) 0.6 0.4 13 45 0 0]
ctp8_2329 -> [980 275 2 2 2 2 (0 1) (0 10) 0.6 0.4 13 45 0 0]
ctp8_4359 -> [1940 325 2 2 2 2 (0 1) (0 10) 0.7 0.5 13 45 0 0]

2- Worst evaluations:

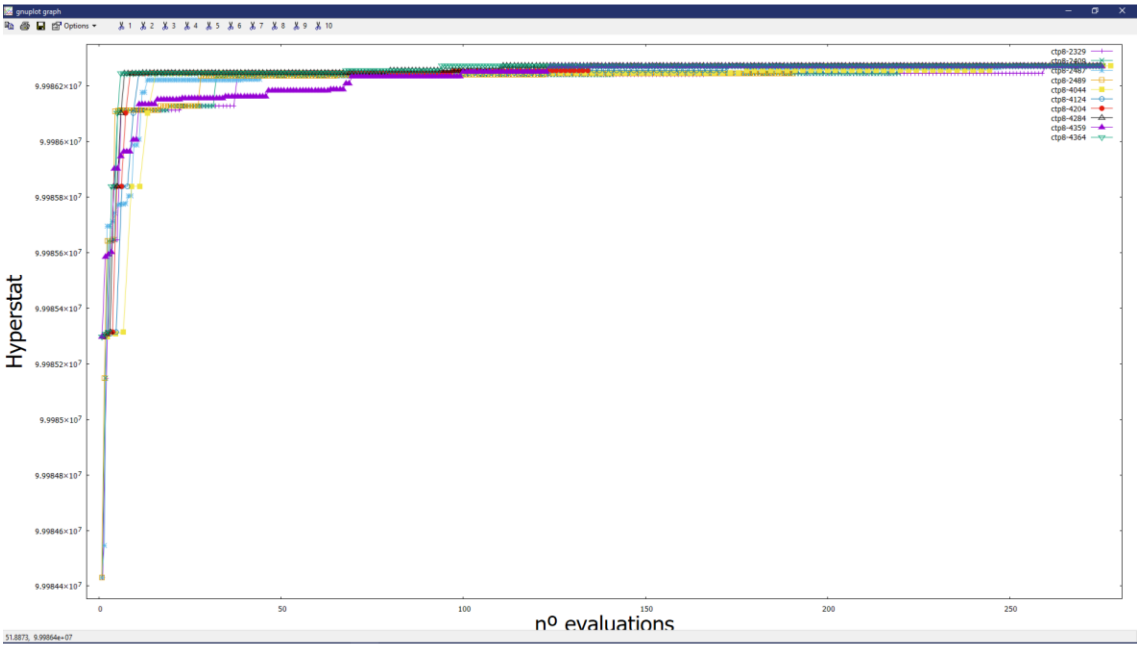
Hyperstat	Input File
(99985141.014467,	ctp8_10)
(99985112.341654,	ctp8_33)
(99985083.407344,	ctp8_37)
(99985001.548864,	ctp8_32)
(99984961.329343,	ctp8_38)
(99984936.395215,	ctp8_1)
(99984914.33782,	ctp8_24)
(99984892.410444,	ctp8_20)
(99984749.636958,	ctp8_43)

Parameter set worse evaluations:

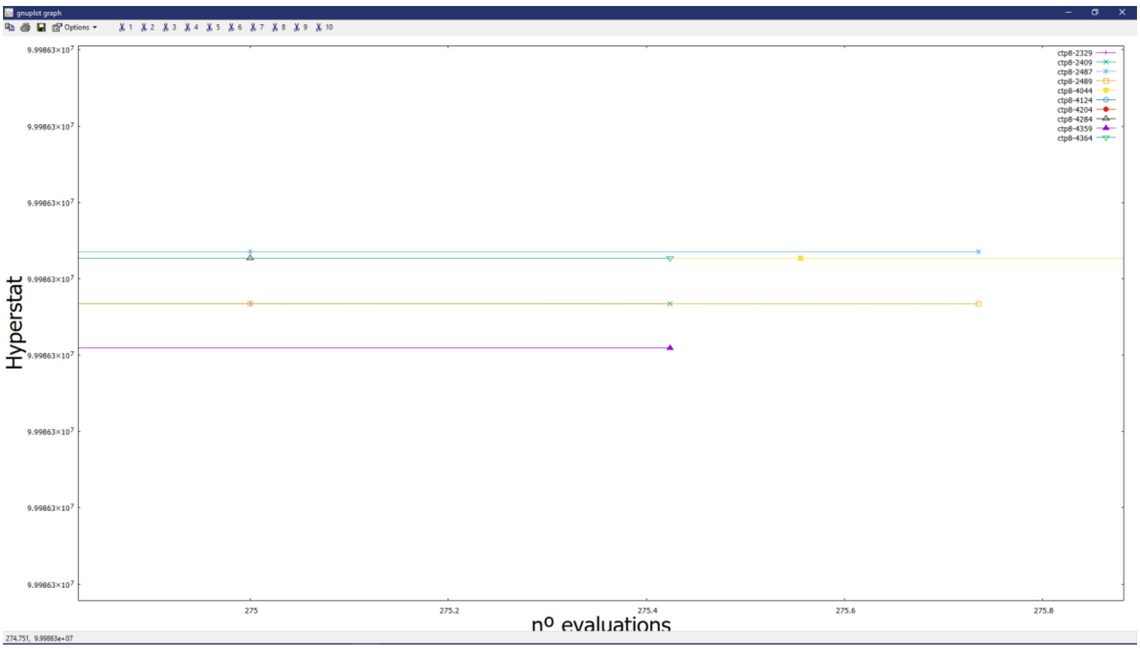
ctp8_10 -> [20 25 2 2 2 2 (0 1) (0 10) 0.8 0.4 5 35 0 0]
ctp8_33 -> [20 25 2 2 2 2 (0 1) (0 10) 0.7 0.5 5 35 0 0]
ctp8_37 -> [20 25 2 2 2 2 (0 1) (0 10) 0.7 0.5 13 25 0 0]
ctp8_32 -> [20 25 2 2 2 2 (0 1) (0 10) 0.7 0.5 5 25 0 0]
ctp8_38 -> [20 25 2 2 2 2 (0 1) (0 10) 0.7 0.5 13 35 0 0]
ctp8_1 -> [20 25 2 2 2 2 (0 1) (0 10) 0.6 0.4 5 15 0 0]
ctp8_24 -> [20 25 2 2 2 2 (0 1) (0 10) 0.7 0.4 5 45 0 0]
ctp8_20 -> [20 25 2 2 2 2 (0 1) (0 10) 0.7 0.4 5 5 0 0]
ctp8_43 -> [20 25 2 2 2 2 (0 1) (0 10) 0.8 0.4 5 35 0 0]

Fifth, for the visualization of the results **GNUPLOT** has been used and the hyper-stats and last-population variables have been plotted resulting in the following figures:

Best parameters election:



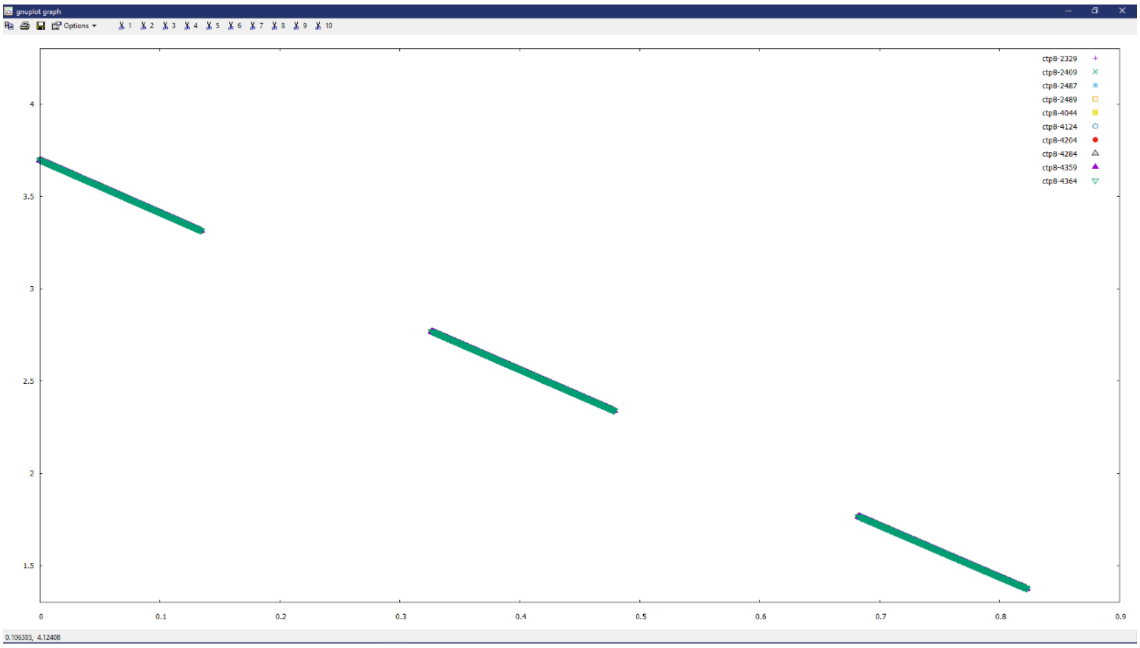
In the following graph we observe how the different configurations quickly (in less than 10 iterations) reach 90% of the optimality achieved by the best configuration. Moreover, in the zoom of the



final area of the metric we can see how the difference between the results is less than 1e-8, so we can conclude that these configurations are in a similar degree of exploitation. In terms of exploration, we have among the top10 a range of (375, 125) and a population range of (980, 1940).

Within the top 10, the 4044 configuration is proposed as the best as it has the best balance between exploration and exploitation being 1e-9 away from the best result (2487), and with a third of the evaluations, although it requires a larger population. For future studies we would start from this configuration, as it has the highest possibility of improvement due to the lower scanning.

As for the visualization of the last population, it has been decided to

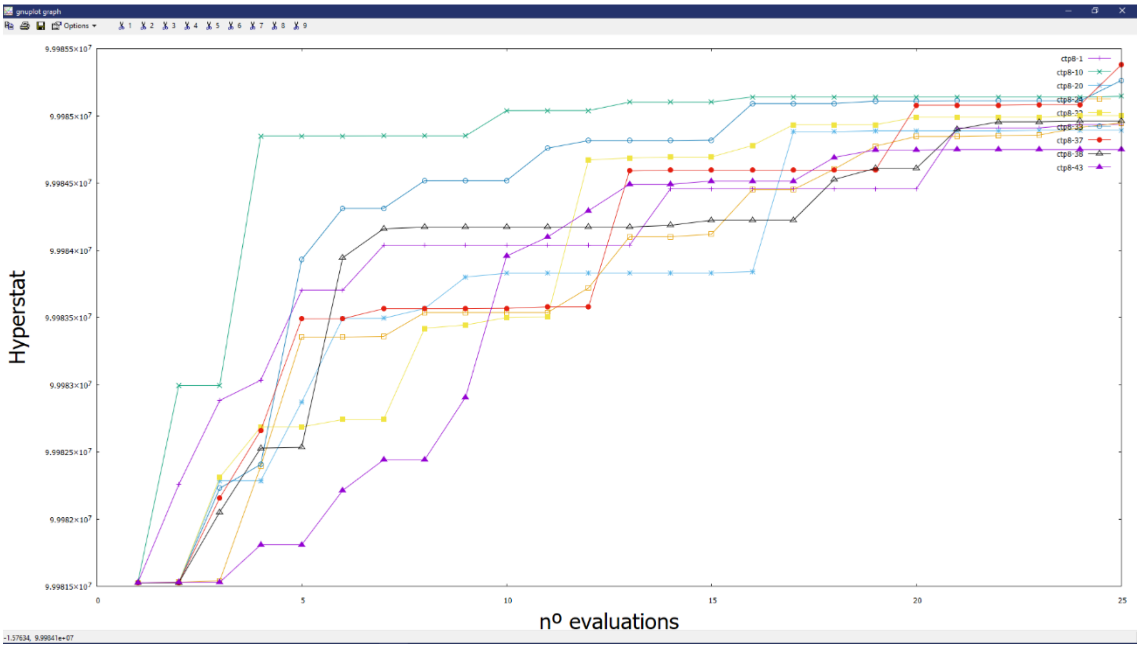


plot it, both as a measure of verification of the characteristics of the solutions, and to observe whether disparate solutions are obtained with similar metrics. In this case it is observed that the top 10 solutions are concentrated in the same area.

Worst parameters election:

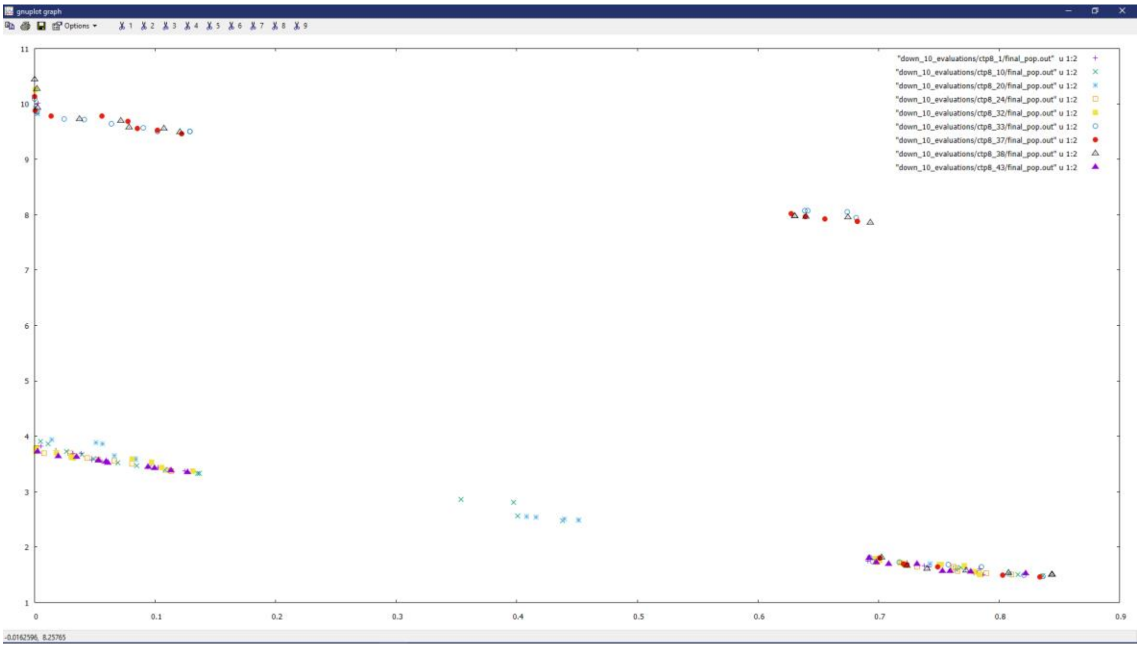
Once the best results have been plotted, the worst ones are explored to see the range of improvement obtained with the choice of parameters.

As we can see in the hyper-stats graph it is reasonable to think that



it has not been possible to reach a steady state due to the low number of evaluations (25 iterations), also configurations with upward trends are observed at the end of the metric evaluation.

As for the populations, this occasion shows different solutions,

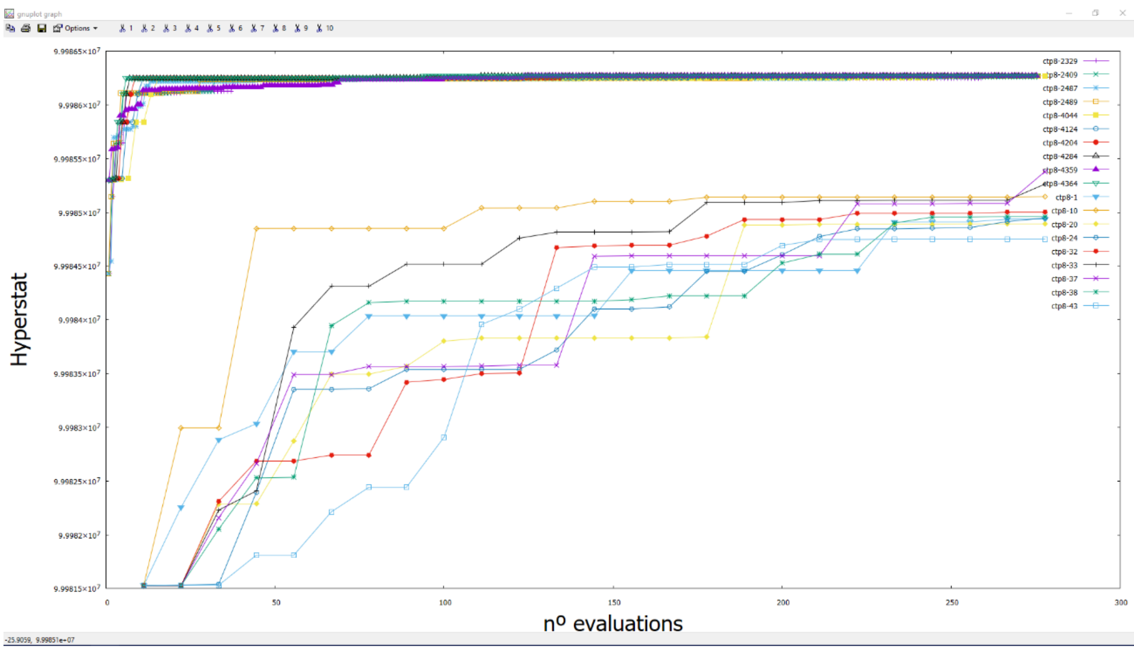


probably because the algorithm has not had enough exploration to really adjust the solution.

Comparison parameters election:

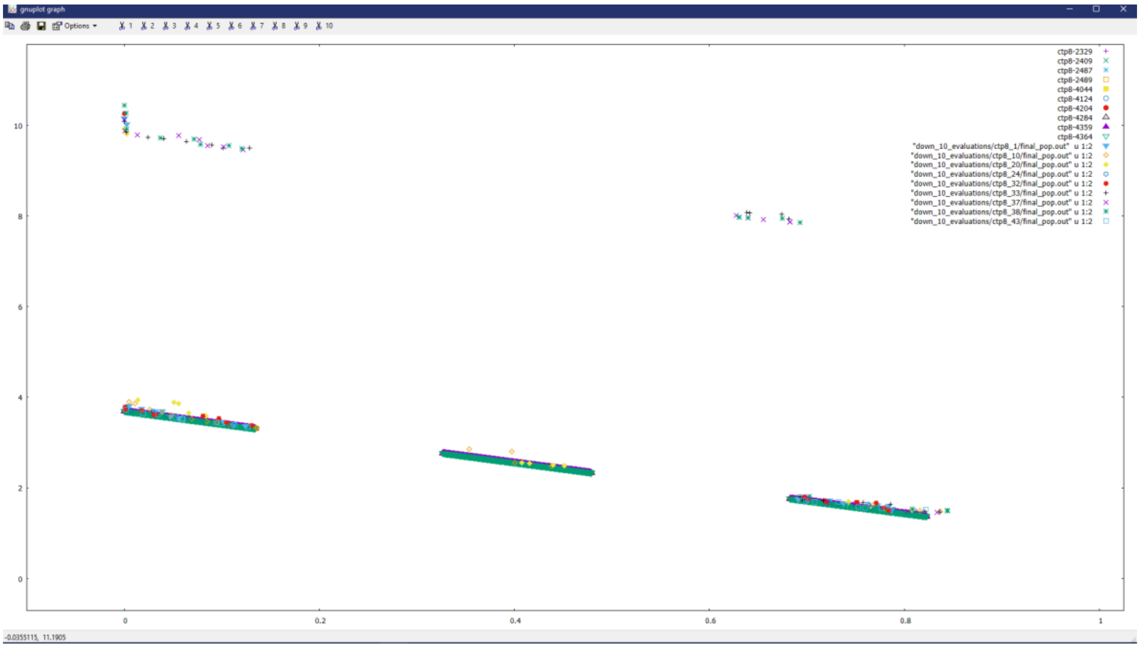
As a purely investigative measure, the best and worst results have been plotted to observe the range of solutions explored.

As for the comparison of the hyper-stats evaluation, we observe



that the solutions have an order of magnitude of difference, besides the lack of evaluations of the worst cases is evident, seeing the density of points of their evaluation.

In terms of population, we also observe that in the worst cases,



most of the population is not close to the solution, and the populations of the best cases are practically overlapping.

* All these results as well as the repository that contains the modifications and the results can be found at:

https://github.com/qwerteleven/DOI_NSGAII

**As for the results of the populations we have kept both the last and the best, for the top10 cases, since the trace of the populations contains the most information, partly due to its human readable characteristic.