Universiteit
Leiden
The Netherlands

Pratical Assignment 2

General Info

- Deadline
  - Submission: December 7, 23:59
  - Submit your report/codes on Blackboard
- Penalty : -0.5/week
- Bonus: 0.5/group for the highest PA grades
- Can work with a group(max. 2 persons)
  - Register your team on Blackboard

General Info

- How to evaluate your PA? Average grade over all the sections.
  - Following the guidelines (sec.1)
  - Algorithm level (sec.2)
  - Code Reproducibility (sec.3)
  - Experimental Results (sec.4)
  - Presentation (sec.5)
  - Code (sec.6)
  - Overall impression (sec.7)
- Formula:
  - $\left(\frac{1}{7}\sum_{i=1}^{7}\text{sec}[i]\right) + penalty + bonus$
- Other:
  - Plagiarism check: if report copies more than 30%, PA grade is 0.

Black-Box Optimization

- Function $f(x)$: $\mathbb{R}^n \rightarrow \mathbb{R}$
- The analytic form is unknown, only $f(x)$ can be evaluate

Using Evolution Strategy to solve black box functions

Five black box functions

- $f(\boldsymbol{x}) \rightarrow \min, \boldsymbol{x}\ \text{in}\mathbb{R}^n$
  - $-100 < x_i < 100$, for all i in {1, …, N}
  - N = 30
- Black-box problems can be found in PA2.zip, to be downloaded from blackboard.

Name

- bbf1.m
- bbf1.mat
- bbf2.m
- bbf2.mat
- bbf3.m
- bbf3.mat
- bbf4.m
- bbf4.mat
- bbf5.m
- bbf5.mat
- test.m

Details

```
>> x = rand(1,30) * 200 - 100

x =

  Columns 1 through 15

   77.3024  -94.2652   -2.0197  -66.4146   95.7361   42.5389    0.0943

  Columns 16 through 30

   63.5094   44.4879  -70.0269   31.9211    3.7190   94.5949   29.7983

>> bbf1(x)

ans =

   1.2494e+05

>> bbf2(x)

ans =

   6.2058e+05

>> bbf3(x)

ans =

   1.0616e+09

>> bbf4(x)

ans =

  125.9323

>> bbf5(x)

ans =
```

Example settings

- Dimensionality: 30
- Lowerbounds and upperbounds: -100, 100
- Multiple independent runs: 20
- Evaluation budget: 10000

- To run your ES on *bbf1* with N=30, lb=$[-100]^N$, ub = $[100]^N$, and a budget of 10,000 function evaluations, you type

```
    [xopt, fopt] =
studentnumber1_studentnumber2_es(@bbf1, 30,
-100 * ones(1,30), 100 * ones(1,30), 10000)
```

- The *bbf* functions expect an N-dimensional row vector x as input (i.e., a vector with a single row and N columns).

What to submit

- A **runnable** MATLAB code, containing your implementation of the algorithm
- A **readable** report that summaries how you implement it, followed a simple experimental comparison.

MATLAB code

- Your ES implementation should consist of **a single** .m file named ***studentnumber1_studentnumber2*_es.m** (replacing studentnumber1 and studentnumber2 by your own studentnumbers) and should be structured as follows:

```
function [xopt, fopt] =
studentnumber1_studentnumber2_es(@function, dimension,
lowerbounds,upperbounds, budget)
…
end
```

What to report

- Final solution quality after 10,000 function evaluations, averaged over 20 runs

	ES configuration A		ES configuration B	
Benchmark	Avg	Std dev	Avg	Std dev
1				
2				
3				
4				
5				

What to report

- Convergence rate plot:
  - ***Optional***: Comparison among different parameter settings, e.g., population size, selection scheme…

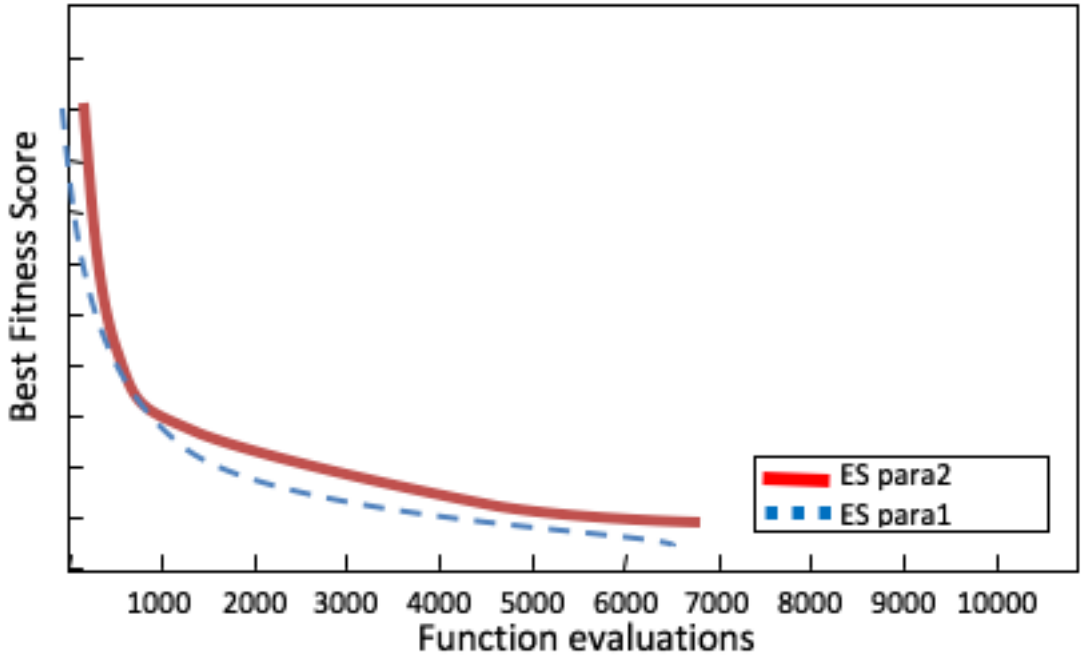

Figure 1: Convergence plot using 10,000 function evaluations, averaged over 20 runs