goa
(Global Optimization Animations)
Optimization Methods - Global Optimization Project

Lorenzo Palloni

University of Florence

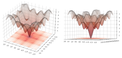
lorenzo.palloni@stud.unifi.it

October 12, 2021

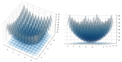
Introduction

What is **goa** (Global Optimization Animations)?

- **goa** is a Python package that implements:
  1. some problems (optimization test functions)
     - Ackley function

     - Rastrigin function

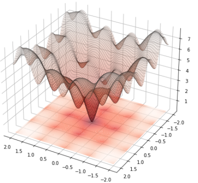
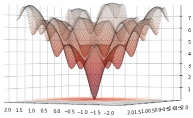
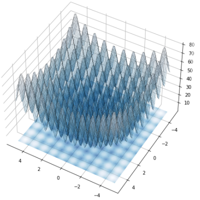
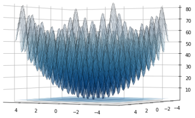
       

  2. some solutions (optimization algorithms)
     - Differential Evolution [1]
     - Memetic Differential Evolution [3]
     - Coordinate Method with Simple Descent Direction [5]
- **goa** aims to animate the execution of these algorithms on a given problem

Ackley and Rastrigin functions

Ackley and Rastrigin functions have common similarities:

- multimodal ($f : \mathbb{R}^n \to \mathbb{R}$)
- many local minima
- unique global minimum at $f(x^\star) = 0$, with $x^\star = (0, \ldots, 0)$

Differential Evolution algorithm

Given a population of p vectors $x_i \in \mathbb{R}^n, i \in \{0, \ldots, p-1\}$,
a high level description of one iteration of the algorithm is the following:

- for each x_i:
  1. apply to x_i a linear transformation and call it *Trial*
  2. restore randomly chosen dimensions of *Trial*
  3. if $f(\mathit{Trial}) < f(x_i)$, then set $x_i := \mathit{Trial}$

Memetic Differential Evolution *vs* Differential Evolution

The two algorithms differ only by the 10^{th} line of code.

Data: $F \in (0,2)$, $CR \in [0,1]$

1 **foreach** $i \in 1, \ldots, p$ **do**
2   let $ii := \mathcal{U}(1, \ldots, n)$;
3   randomly choose $k_1, k_2, k_3 \in \{1, \ldots, p\} \setminus \{i\}$;
4   let $Trial := x_{k_1} + F(x_{k_2} - x_{k_3})$;
5   **for** $j \in 1, \ldots, n : j \neq ii$ **do**
6   **if** $\mathcal{U}(0,1) < CR$ **then**
7   let $Trial^{(j)} := x_i^{(j)}$;
8   **end**
9   **end**
10   let $Trial := \mathcal{L}(f, Trial)$;
11   **if** $f(Trial) < f(x_i)$ **then**
12   let $x_i := Trial$;
13   **end**
14 **end**

Algorithm 1: Memetic Differential Evolution

Data: $F \in (0,2)$, $CR \in [0,1]$

1 **foreach** $i \in 1, \ldots, p$ **do**
2   let $ii := \mathcal{U}(1, \ldots, n)$;
3   randomly choose $k_1, k_2, k_3 \in \{1, \ldots, p\} \setminus \{i\}$;
4   let $Trial := x_{k_1} + F(x_{k_2} - x_{k_3})$;
5   **for** $j \in 1, \ldots, n : j \neq ii$ **do**
6   **if** $\mathcal{U}(0,1) < CR$ **then**
7   let $Trial^{(j)} := x_i^{(j)}$;
8   **end**
9   **end**
10   // let $Trial := \mathcal{L}(f, Trial)$
11   **if** $f(Trial) < f(x_i)$ **then**
12   let $x_i := Trial$;
13   **end**
14 **end**

Algorithm 2: Differential Evolution

Animations - DE vs MDE on Ackley

- Left example: DE reaches convergence in 38 iterations
- Right example: MDE reaches convergence in 3 iterations

DE on Ackley

MDE on Ackley

Animations - DE vs MDE on Quadratic

- Left example: DE reaches convergence in 19 iterations
- Right example: MDE reaches convergence in 1 iteration

DE on Quadratic

MDE on Quadratic

Command line interface (CLI)

goa provides a CLI that allows to choose:

1. problem
2. problem bounds (optional)
3. optimization algorithm
4. local optimizer (required only for memetic global optimizer)
5. animation filepath (optional)

```
(goa) ➜  presentation git:(main) ✗ python -m goa
Select a problem (Ackley, Rastrigin, quadratic) [Ackley]:
[OPTIONAL] Change the problem bounds [(-2.5, 2.5)]:
Select an optimization algorithm (MDE, DE, CM) [DE]:
[REQUIRED only with MDE] Select a local search algorithm (CM, None) [None]:
[OPTIONAL] Want an animation? Provide a filepath .gif []: 'example.gif'
Iteration:    5 | RMSE: 0.91623379
Iteration:   10 | RMSE: 0.84967181
Iteration:   15 | RMSE: 0.74277798
Iteration:   20 | RMSE: 0.06209373
Iteration:   25 | RMSE: 0.00837453
Iteration:   30 | RMSE: 0.00156031
Iteration:   35 | RMSE: 0.00026433
Terminated at Iteration: 37
```

Conclusion

Who would need **goa**?

- anyone that would like to support an explanation
- anyone that would like to have another point of view

What could be improved?

- scalability (abstract classes for both global and local optimization algorithms)
- reproducibility (all RNGs should depend on a single random seed)

Where can the implementation be found?

- *https://github.com/deeplego/goa*

Thanks for your attention!

Do you have any questions?

References

- Storn, R. and Price, K., 1997. Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. Journal of global optimization, 11(4), pp.341-359.
- Locatelli, M., Maischberger, M. and Schoen, F., 2014. Differential evolution methods based on local searches. Computers & operations research, 43, pp.169-180.
- Mansueto, P. and Schoen, F., 2021. Memetic differential evolution methods for clustering problems. Pattern Recognition, 114, p.107849.
- Kononova, A.V., Caraffini, F. and Bäck, T., 2020. Differential evolution outside the box. arXiv preprint arXiv:2004.10489.
- Grippo, L. and Sciandrone, M., 2011. Metodi di ottimizzazione non vincolata. Springer Science & Business Media.

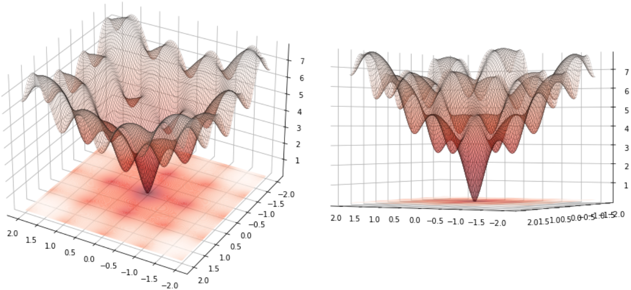
Backup slides

Definition (Ackley function)

$$f(x_1 \cdots x_n) = -20exp(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}) - exp(\frac{1}{n}\sum_{i=1}^{n} cos(2\pi x_i)) + 20 + e$$

$$-32 \leq x_i \leq 32$$

minimum at $f(x^\star) = 0$, with $x^\star = (0, \ldots, 0)$



Definition (Rastrigin function)

$$f(x_1 \cdots x_n) = 10n + \sum_{i=1}^{n} (x_i^2 - 10cos(2\pi x_i))$$

$$-5.12 \leq x_i \leq 5.12$$

minimum at $f(x^\star) = 0$, with $x^\star = (0, \ldots, 0)$