FEDERAL STATE AUTONOMOUS EDUCATIONAL INSTITUTION
OF HIGHER EDUCATION
ITMO UNIVERSITY

Report
on the practical task No. 4
Algorithms for unconstrained nonlinear optimization. Stochastic and metaheuristic algorithms.

Performed by:
Roman Bezaev
J4133c

Accepted by:
Dr Petr Chunaev

St. Petersburg
2021

1 Goal

The use of stochastic and metaheuristic algorithms (Simulated Annealing, Differential Evolution, Particle Swarm Optimization) in the tasks of unconstrained nonlinear optimization and the experimental comparison of them with Nelder-Mead and Levenberg-Marquardt algorithms.

2 Formulation of the problem

Task I.
Generate the noisy data (x_k, y_k) where $k = 0, \ldots, 1000$, according to the rule:

$$y_k = \begin{cases} -100 + \delta_k, & f(x_k) < -100, \\ f(x_K) + \delta_k, & -100 \leq f(x_k) \leq 100, \ x_k = \frac{3k}{1000} \\ 100 + \delta_k, & f(x_k) > 100 \end{cases}$$

$$f(x) = \frac{1}{x^2 - 3x + 2} \tag{1}$$

where $\delta_k \sim N(0,1)$ are values of a random variable with standard normal distribution. Approximate the data by the rational function

$$F(x, a, b, c, d) = \frac{ax + b}{x^2 + cx + d} \tag{2}$$

by means of least squares through the numerical minimization of the following function:

$$D(a, b, c, d) = \sum_{k=0}^{1000} (F(x_k, a, b, c, d) - y_k)^2) \tag{3}$$

To solve the minimization problem, use Nelder-Mead algorithm, Levenberg-Marquardt algorithm and at least two of the methods among Simulated Annealing, Differential Evolution and Particle Swarm Optimization. If necessary, set the initial approximations and other parameters of the methods. Use $\varepsilon = 0.001$ as the precision; at most 1000 iterations are allowed. Visualize the data and the approximants obtained in a single plot. Analyze and compare the results obtained (in terms of number of iterations, precision, number of function evaluations, etc.).

Task II.
Choose at least 15 cities in the world having land transport connections between them. Calculate the distance matrix for them and then apply the Simulated Annealing method to solve the corresponding Travelling Salesman Problem. Visualize the results at the first and the last iteration.

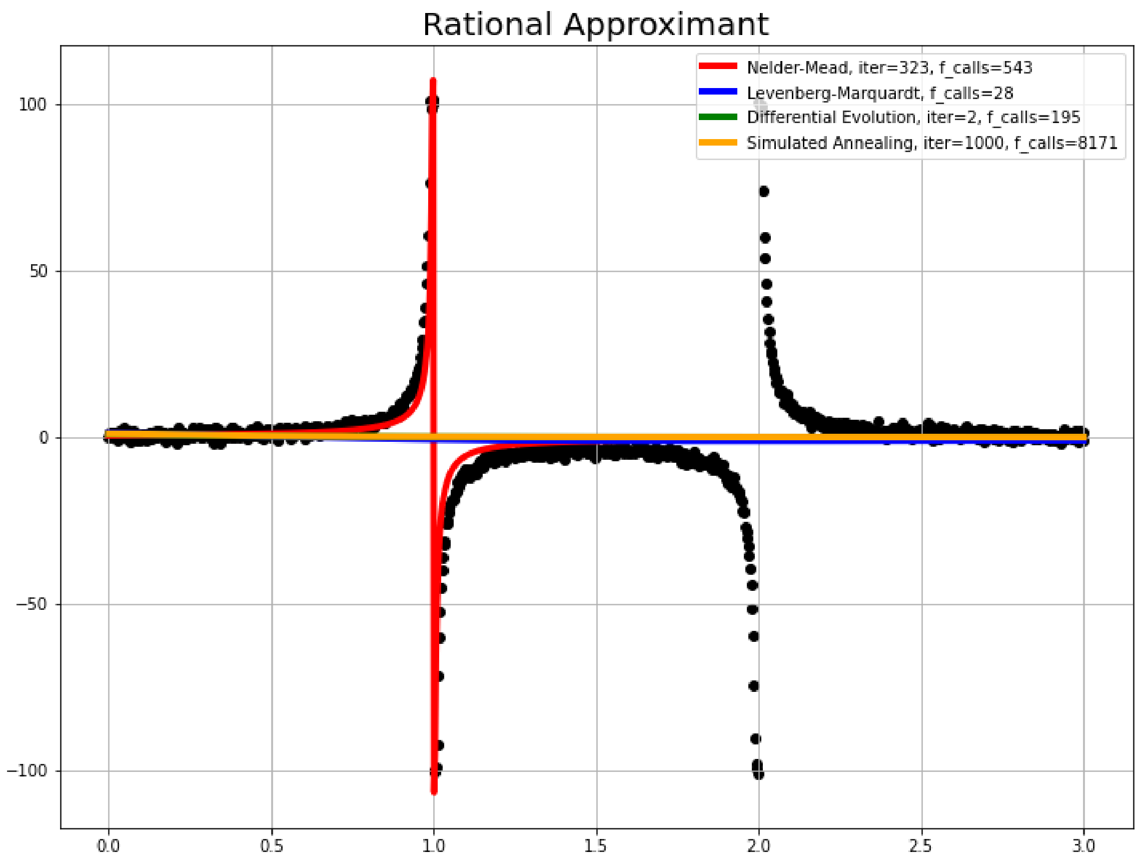
3 Brief theoretical part

Simulated annealing (SA) is a probabilistic technique for approximating the global optimum of a given function. Specifically, it is a metaheuristic to approximate global optimization in a large search space for an optimization problem. It is often used when the search space is discrete. For problems where finding an approximate global optimum is more important than finding a precise local optimum in a fixed amount of time, simulated annealing may be preferable to exact algorithms such as gradient descent, Branch and Bound. The name of the algorithm comes from annealing in metallurgy, a technique involving heating and controlled cooling of a material. The state of some physical systems, and the function E(s) to be minimized, is analogous to the internal energy of the system in that state. The goal is to bring the system, from an arbitrary initial state, to a state with the minimum possible energy. The principle of simulated annealing can be expressed in the following algorithm. At each step, the simulated annealing heuristic considers some neighboring state of the current state, and probabilistically decides between moving the system to that state or staying instate. These probabilities ultimately lead the system to move to states of lower energy. Typically this step is repeated until the system reaches a state that is good enough for the application, or until a given computation budget has been exhausted.

Differential evolution (DE) is a method that optimizes a problem by iteratively trying to improve a candidate solution with regard to a given measure of quality. It is meta-heuristic algorithm, and it makes no assumptions about the problem being optimized and can search very large spaces of candidate solutions. However, it it not guaranteed that an optimal solution will be ever found. DE optimizes a problem by maintaining a population of candidate solutions and creating new candidate solutions by combining existing ones according to its simple formula, and then keeping whichever candidate solution has the best score or fitness on the optimization problem at hand. In this way the optimization problem is treated as a black box that merely provides a measure of quality given a candidate solution and the gradient is therefore not needed.

4 Results

Task I.



Only the Nelder-Mead method has found break of the function. Other methods approximated the function by an almost straight line, which leads to large squared error. As we see for meta-heuristic methods this function turned out to be too complex and they cannot approximate it properly. Simulated annealing method also needs a lot of iterations and functions calls. DE method usually need much less iterations and functions calls.

Task II. I chose dataset with 15 abstract cities, for which a solution to the traveling salesman problem is sought. https://people.sc.fsu.edu/~jburkardt/datasets/cities/lau15_xy.txt.

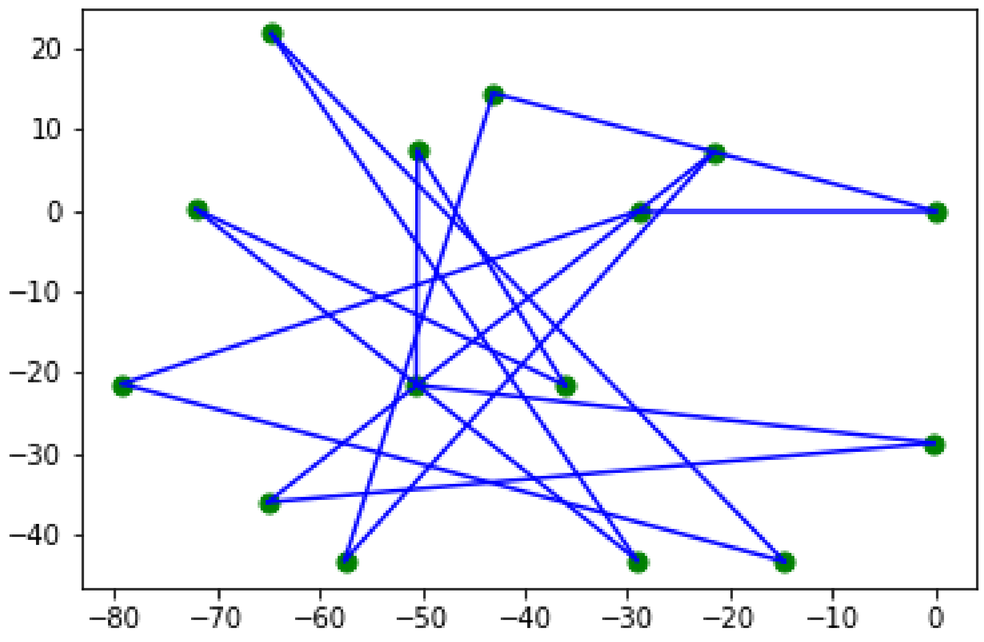


Figure 1: First iteration

On the first iteration route is non optimal. Itinerary that salesman should walk is very long. **Route length: 818.02**

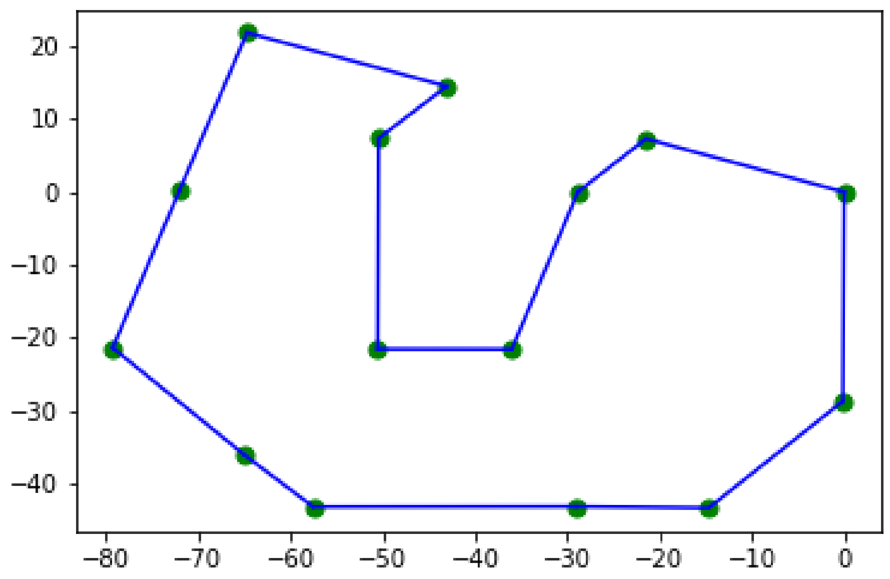


Figure 2: Last iteration

On the last iteration algorithm found the best or one of the best solutions. **Route length: 301.35**

5 Conclusions

The performance of two meta-heuristic methods – differential evolution and simulate annealing were analyzed and compared with that of direct Nelder-Mead algorithm and gradient-based Levenberg-Marquardt algorithm. Both meta-heuristic methods happened to be worse than Nelder-Mead method because of function complexity. Gradient-based methods such as Levenberg-Marqardt show poor results because it not well suited for the problems with non-continuous search space with several optimum points. Travelling salesman problem can be solved by using Simulated Annealing method.

6 Appendix

https://github.com/trixdade/ITMO_algorithms/blob/master/Algorithms_Lab4.ipynb