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

Rami Al-Naim

J4132c

Accepted by

Dr Petr Chunaev

St. Petersburg
2020

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

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

$$y_k = \begin{cases} -100 + \sigma_k, \ f(x_k) < -100 \\ f(x) + \sigma_k, \ \ -100 \leq f(x_k) \leq 100 \\ 100 + \sigma_k, \ f(x_k) > -100 \end{cases}$$

$$x_k = \frac{3k}{1000}, \ k = 1...1000$$

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

$$\sigma_k = N(0, 1)$$

Approximate the data by the rational function

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

by means of least squares through the numerical minimization.

3 Brief theoretical part

During this work four methods of optimization: Nelder-Mead algorithm, Levenberg-Marquardt algorithm, Simulated Annealing, Particle Swarm Optimization.

Nelder-Mead algorithm is a direct heuristic method which uses simplexes to minimize the function. The Nelder-Mead algorithm's implementation from `scipy` module were used.

Levenberg-Marquardt algorithm is a second-order optimization method which minimizes the sum of squared errors and widely used in problems of multidimensional curve-fitting problem. The Levenberg-Marquardt algorithm's implementation from `scipy` module were used.

Simulated Annealing is a metaheuristic algorithm that solves the optimization problem similar to the process of annealing in metallurgy. The decision on changing the value of an argument depends on the temperature function. The Simulated Annealing algorithm's implementation from `scipy` module were used.

Particle Swarm Optimization is a bio-inspired algorithm that simulates the movement of possible solutions (particles). The movement of the particles depends on the hyperparametes, best position found by the whole swarm and best position found by the particle. The Particle Swarm Optimization algorithm's implementation from `pyswarm` module were used.

4 Results

All algorithms were used to optimize the same noised data. The the statistics of mean squared error, number of iterations and elapsed time were collected and presented in Table 1. The results of approximated data is present in the Figure 1.

Algorithm	MSE	# of iteration	Elapsed time, ms
Simulated Annealing	136.061639	61	16000.0
Swarm optimization	154.888025	1000	4900.0
Nelder-Mead	239.141254	217	65.2
Levenberg-Marquardt	136.065805	44	30.1

Table 1: Comparison of different optimization methods

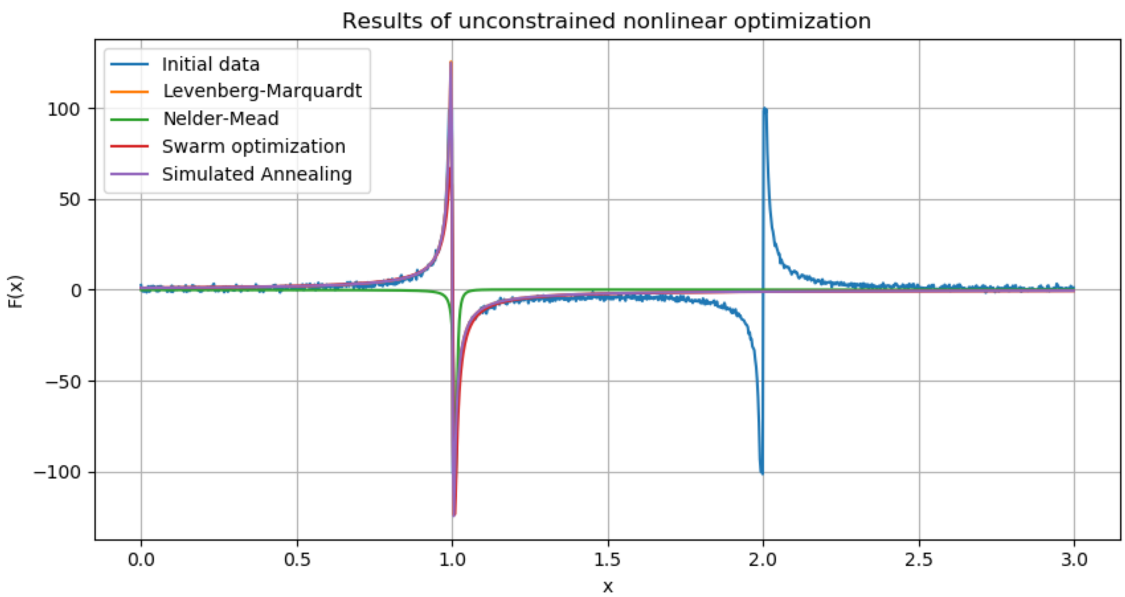

Figure 1: Results of optimization

5 Conclusions

During this work metaheuristic algorithms (Simulated Annealing and Swarm Optimization) were compared with direct and second-order optimization algorithms

(Nelder-Mead and Levenberg-Marquardt).

From the Figure 1 it can be seen what Simulated Annealing, Swarm Optimization and Levenberg-Marquardt methods all found local minimum and well approximated only one discontinuity (at $x = 1$). At the same time Nelder-Mead, the only direct method visibly showed worse results (green line on the plot).

Table 1 shows that Simulated Annealing, Swarm Optimization and Levenberg-Marquardt provided roughty the same mean square error, and Nelder-Mead showed the leats accurate result. However, among all the considered methods Levenberg-Marquardt showed the best results since it took the least elapsed time, small number of iterations and decent accuracy, whereas Swarm optimization method used 1000 iterations and nearly 5 seconds and Simulated Annealing gave the best accuracy at cost of 16 seconds of runtime.

Appendix

The Python code for this task can be found here: GitHub.