Krzysztof CHWASTEK, Mariusz NAJGEBAUER, Jan SZCZYGŁOWSKI

Politechnika Częstochowska, Instytut Elektroenergetyki

Performance of some novel optimization algorithms

***Abstract**. Three global optimization algorithms are tested against some chosen benchmark tests with known minima. Particle Swarm Optimization yields results closest to the sought minima, but is also the slowest algorithm. The hybrid Simplex-Simulated Annealing approach requires fine-tuning of its settings in most cases.*

Streszczenie.** Przetestowano zachowanie trzech algorytmów optymalizacji globalnej wykorzystując wybrane zadania testowe o znanych minimach globalnych. Optymalizacja metodą roju cząstek daje rezultaty najbardziej zbliżone do poszukiwanych, lecz równocześnie jest to najwolniejszy algorytm z rozpatrywanych. Podejście hybrydowe simpleks - symulowane wyżarzanie wymaga dostrajania swoich parametrów w większości przypadków. (**Ocena wybranych nowoczesnych algorytmów optymalizacji).

Keywords: optimization, benchmark functions, MATLAB.
Słowa kluczowe: optymalizacja, funkcje testowe, MATLAB.

Introduction

Optimization problems arise in many problems in electrical engineering, e.g. parameter estimation for hysteresis models [1-3], design of electric machines [4,5] and other devices [6-8] *etc*. Classical methods used for solving highly nonlinear problems are sometimes slow convergent or not robust enough, therefore much attention is paid to alternative approaches based e.g. on artificial intelligence methods.

In order to compare the performance of optimization algorithms, a number of benchmark problems with known solutions has been proposed [9-12]. A similar approach is applied in cryptography to test the strength of developed ciphers. The aim of the present paper is to provide a comparison of some novel optimization algorithms. Particular attention is paid to the algorithm, which mimicks a social-behavioural collective action, i.e. Particle Swarm Optimization [12-14]. Collective interaction between similar units is well recognized e.g. in ferromagnetism [15], where it has been transformed into a very successful concept of “effective field” [16]. This century-old idea has become the cornerstone of many contemporary descriptions of magnetization phenomenon. An interesting association of the idea of collective interaction with the concepts advanced by such renowned scientists and philosophers as Isaak Newton, Thomas Hobbes, Adam Smith and John Locke, as well as by the 20th century economists, has been presented in Ref. [12].

Another interesting association between contemporary condensed matter physics and optimization theory is behind the Simulated Annealing concept [17]. In statistical mechanics the behaviour of large aggregates of atoms subject to cooling is highly influenced by the cooling rate. Slow decrease of sample temperature results in smearing out the structural disorder and favours the approach to an equillibrium state corresponding to the global energy minimum. On the other hand, rapid cooling (quenching) yields defects and glass-like intrusions inside the material. It leads to the final state with a higher energy level than previously considered. This phenomenon is particularly well known to technologists working on amorphous and nanocrystalline alloys [18]. Simulated Annealing method developed by N. Metropolis *et al.* [19] works on a similar principle. Starting from a given point x_0 in the search space, a new candidate point x_1 is chosen according to some criterion and the energy (cost) values are calculated for both points. If $\Delta E = E(x_1) - E(x_0) \le 0$ then the point x_1 is accepted and becomes the starting point for the next iteration, otherwise the new point is accepted, but conditionally, only if $\exp(-\Delta E / T) \ge r$, where *r* is a random number from the unit interval and T is a parameter called temperature. Initially the temperature value is kept high, but it is decreased during successive iterations. In the paper an implementation of the hybrid approach, which combines the Simulated Annealing algorithm with the nonlinear simplex method, is used [20,21].

The third optimization method considered in the paper (Shuffled Complex Evolution Approach) is based on a synthesis of four concepts that have proved to be successful in global optimization i.e. combination of probabilistic and deterministic approaches, clustering, systematic evolution of a complex of points spanning the space in the direction of global improvement and competitive evolution [20,22].

Selected benchmark problems

As pointed out in the previous section, there exists a number of benchmark tasks with known solutions used for testing the robustness of optimization algorithms. Those may be either unimodal (having just one global minimum) or multimodal. We have chosen some less known benchmark functions from the set prepared by R. Oldehuis [11]. The Figures presented in the paper were obtained using the *ezimage* tool developed by this author.

The first considered benchmark was the unimodal Matyas function, given with the relationship

$$f(x_1, x_2) = 0{,}26\left(x_1^2 + x_2^2\right) - 0{,}48\, x_1\, x_2 \qquad (1)$$

The function has a global minimum equal to zero at $(x_1, x_2) = (0{,}0)$. The shape of the Matyas function is shown in Figure 1. White point denotes the global minimum.

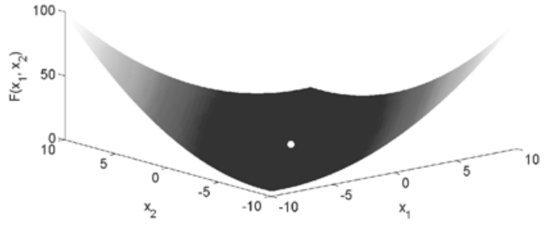

Fig.1. The shape of the Matyas function

The second unimodal function under consideration was the Levi13 function. Its formula was a bit more complex:

$$\begin{aligned} f(x_1, x_2) &= \sin(3\pi\, x_1)^2 + (x_1 - 1)^2\left(1 + \sin(3\pi\, x_2)^2\right) + \ldots \\ &+ (x_2 - 1)^2\left(1 + \sin(2\pi\, x_2)^2\right) \end{aligned} \qquad (2)$$

The function has a global minimum equal to zero at $(x_1, x_2) = (1{,}1)$. The function is shown in Figure 2.

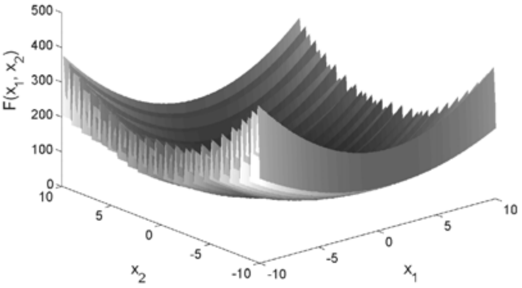

Fig. 2. Levi13 function

As the example of a multimodal function we have chosen the Himmelblau function, depicted in Figure 3.

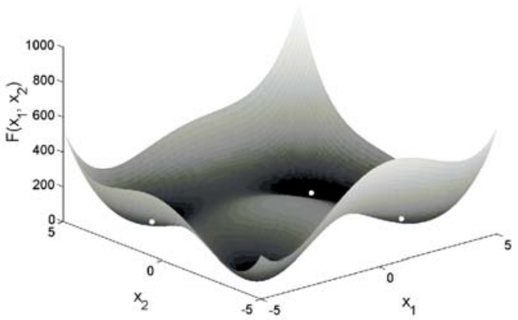

Fig. 3. Himmelblau function (white points denote global minima)

Its formula is given with the following relationship:

$$f(x_1, x_2) = \left(x_1^2 + x_2 - 11\right)^2 + \left(x_1 + x_2^2 - 7\right)^2 \tag{1}$$

The function has four equal minima equal to zero for the following coordinates:

$(x_1, x_2) = (3;2)$;

$(x_1, x_2) = (-2{,}805118; 3{,}131312)$;

$(x_1, x_2) = (-3{,}779310; -3{,}283186)$;

$(x_1, x_2) = (3{,}584428; -1{,}848126)$.

The locations of all the minima can be found analytically, but the expressions are long and complicated.

Modelling results

We have used the Matlab codes developed by B. Donckels [20] for all tests. The calling convention for all algorithms was quite similar. In the first attempt we have kept the default settings for the optimization routines, as proposed by the author. These settings were chosen arbitrarily basing on experience of different researchers, but they may be easily fine-tuned for specific needs, using scripts provided by the code developer. Settings common for all algorithms are given in Table 1 below.

Table 1. Some settings common for all considered algorithms

Setting	Value
Maximum number of iterations	2500
Maximum number of function evaluations	2500
Maximum difference between the coordinates of the vertices	10^{-3}

The lower and upper bounds for the variables in the search space were preset for each considered benchmark function. Within the search space a vector of guess solutions (actually a two dimensional matrix), consisting of 100 elements, was randomly generated and saved for further use. The codes provided by B. Donckels require some guess solution values for initialization. The same vectors of guess solutions were used for all optimization routines working on a specific benchmark function in order to avoid side-effects. The point coordinates and the cost values, obtained with the optimization routines, were saved in appropriate Matlab matrices. On the basis of carried out tests and the analysis of profiler codes a general conclusion about the performance of the algorithms may be drawn. Particle Swarm Optimization (with contraction factor introduced to avoid premature convergence [14]) is the slowest routine, however it yields most repeatitive results, which are moreover closest to the known solution. Shuffled Complex Evolution approach is a good compromise in terms of speed and accuracy. The hybrid Simplex-Simulated Annealing algorithm is most probably the fastest one in the considered group, but in some cases fails to find the optimal solutions for its default settings. This effect is clearly visible in 3D representation of solutions obtained for the Levi13 function, depicted in Figure 4.

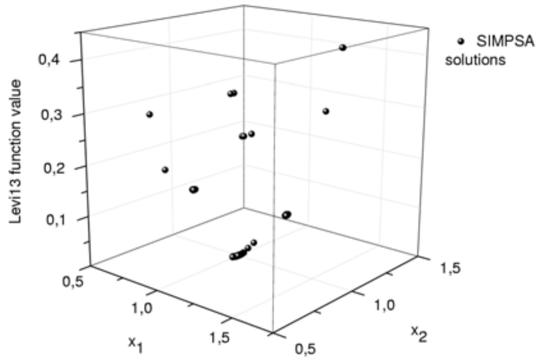

Fig. 4. Solutions of SIMPSA test run on Levi13 function

In nine cases the differences between the expected value 0 and final fitness values differed more than 0,25 (in the Figure only five such points are visible, because in some cases the solutions have converged to the same points). In the worst case the final value returned by the algorithm was 0,4412 and the minimum was found at (1,6603; 0,9937), which by any means cannot be satisfactory. We suspected that this effect might be due to some non-optimal algorithm settings, e.g. too fast cooling rate. Therefore we carried out additional tests of the algorithm with altered settings. An overall improvement of the algorithm performance was obtained after modification of a number of its settings: COOL_RATE = 1,1 (the default value is 10, which means faster convergence), MIN_COOLING_FACTOR = 0,95 instead of 0,9, MAX_ITER_TEMP_FIRST and MAX_ITER_TEMP_LAST increased up to 100 (to be compared to 50, which is the default value; the meaning of these parameters is the number of iterations in the preliminary and the final (pure simplex) loop, respectively); MAX_ITER_TEMP was increased from 10 to 50 (this is the number of iterations in the remaining temperature loop). TEMP_START was set to 100 (originally it was predetermined by the code itself). As it can be easily seen, most of the introduced modifications were aimed at increasing the number of iterations within the algorithm loops. The corresponding increase in computation time was about the factor of two, yet the speed of the code was satisfactory (about a minute for the whole 100-element batch on a modern Intel i5 machine running 64-bit Matlab 7.10 on Windows 7. For the modified settings and the same set of guess points the algorithm has stuck just once at a point with fitness value above 0,25 (0,2884 @ (1,3257; 0,7743)). The number of iterations in that case was 522 and the number of function evaluations was 928. In 22 cases the final fitness values were of the order of 0,1, but otherwise they were of the order 10^{-5} – 10^{-4} (the found points were quite close to the expected global minimum).

An interesting case was the multimodal Himmelblau function. All algorithms were able to locate the global minima; as usual PSO was the slowest, but most accurate, whereas SIMPSA revealed the opposite behaviour. In just one case SIMPSA did not approach one of the global minima by less than 0,1 (the value was 0,1026 for the point (3,025; 1,9134), which, however, remained not too far from the first minimum (3;2)).

Conclusions

Performance of three novel global optimization algorithms was tested using chosen benchmark tests with known solutions. The algorithms had to be called several times due to their stochastic operation mode. In all cases the code implementing Particle Swarm Optimization (with constriction factor to prevent premature convergence) gave the solutions closest to the sought minima. However, the preferred procedure is the Shuffled Complex Evolution algorithm, which combines accuracy sufficient for engineering purposes with increased speed. The default settings for the algorithms were satisfactory, however in some cases it was necessary to fine-tune them by hand (cf. “No free lunch theorem” [23]). This was particularly visible for the SIMPSA algorithm, which has failed to converge several times even for the simple *matyas* function.

Generally speaking, all the Matlab codes developed by B. Donckels [20] are useful for engineering applications. The implementations are fast and flexible enough to include them as part of more complex calculation routines.

Appendix – Matlab calling procedure

The functions to be minimized have to be called several times due to stochastic nature of the optimization procedures. The Matlab function *arrayfun* is useful for handling batch function calls.

```
fun = @(x1,x2) PSO(‘matyas’, [x1,x2], [-10,-10], [10,10];
numpoints = 100;
m1 = -10 + 20.*rand(numpoints,1);
m2 = -10 + 20.*rand(numpoints,1);
% the function ‘matyas’ is optimized using PSO in two
% dimensional space <-10;10> x <-10;10> consisting of 100
% points, whose coordinates are randomly generated
[xcoord, fvalue, exitflag, out] = arrayfun(fm, m1, m2,
‘UniformOutput’, false);
```

The results of individual function calls are stored as cell arrays consisting of appropriate structures. These may be post-processed using *cell2mat or cell2struct* procedures.

Acknowledgements

K. Chwastek is grateful for support within the framework of research grant N N510 702540 from National Centre of Science.

REFERENCES

[1] Naghizadeh R.-A., Vahidi B., Hosseinian S. H., Parameter identification of Jiles-Atherton model using SFLA, *COMPEL* 31 (2012), 1293-1309

[2] dos Santos Coelho L., Cocco Mariani V., Leite J. V., Solution of Jiles-Atherton vector hysteresis parameters estimation by modified Differential Evolution strategies, *Expert Syst. Appl.* 39 (2012), 2021-2025

[3] Jianbin Zeng, Baodong Bai, Evaluation of Jiles-Atherton hysteresis model’s parameters using mix of simplex method and Simulated Annealing method, Proc. ICEMS’2008, 17-20.10.2008, Wuhan, China, 4033-4036, INSPEC No. 10467755

[4] Petkovska L., Cvetkovski G., Genetic algorithm coupled with FEM to solve design optimization problem of an inductor, *Przegl. Elektrot.* 12 (2006), 100-103

[5] Marčič T., Štumberger B., Štumberger G., Hadžiselimovič M., Virtič P., Determining parameters of a line-start interior Permanent Magnet Synchronous Motor by the Differential Evolution, *IEEE Trans. Magn.* 44 (2008), 4385-4388

[6] Azzaoui S., Srairi K., El Hachemi Benbouzid M., Non linear magnetic hysteresis modelling by Finite Volume Method for Jiles-Atherton model optimizing by a genetic algorithm, *J. Electromagn. Analys. Appl.* 3 (2011), 191-198

[7] Scaliukh A., Soloviev A., Dmitrieva E., Shevtsova M., An optimal parameters determination for ferroelectric's polarization model, *ASME 2010 10th Biennial Conference on Engineering Systems Design and Analysis*, Volume 2 / Applied Mechanics, paper no. ESDA2010-24841, pp. 263-269, Istanbul July 12-14, 2010, http://dx.doi.org/10.1115/ESDA2010-24841

[8] Cheng M., Marinovic W., Watson M., Ourselin S., Passenger J., De Visser H., Salvado O., Riek S., Abdominal palpation haptic device for colonoscopy simulation using pneumatic control, *IEEE Trans. Haptics*, vol. 5 (2012), 97-108

[9] Floudas C.A., Pardalos P.M., A collection of test problems for constrained global optimization algorithms, *Lecture Notes in Computer Science* 455, Springer-Verlag (1990)

[10] Horst R., Pardalos, P.M. (Eds.), *Handbook of global optimization*, Kluwer (1995)

[11] Oldenhuis R., Many testfunctions for global optimizers, http://www.mathworks.com/matlabcentral/fileexchange/23147-many-testfunctions-for-global-optimizers

[12] Mishra S. K., Munich University, Some new test functions for global optimization and performance of repulsive particle swarm method, *MPRA paper* No. 2718, http://mpra.ub.uni-muenchen.de/2718/, posted 7.11.2007

[13] Kennedy J., Eberhart R., Particle Swarm Optimization, *Proc. IEEE Int. Conf. on Neural Networks*, Perth, Australia, 1995, IV:1942-1948

[14] Clerc M., Kennedy J., The particle swarm-explosion, stability, and convergence in a multidimensional complex space, *IEEE Trans. Evol. Comput.* 6 (2002), 58-73

[15] Vicsek T., A question of scale, *Nature* 411(2001), 421

[16] Weiss P., L’hypothèse du champ moléculaire et la propriété ferromagnétique, *J. de Phys.* 4o série, t. VI (1907), 661-690

[17] Kirkpatrick S., Gelatt C. D., Jr., Vecchi M. P., Optimization by Simulated Annealing, *Science* 220 (1983), 671-679

[18] Davies H. A., Gibbs M. R. J., Amorphous alloys, Chapter 1, Volume 4: Novel materials, in: *Handbook of magnetism and advanced materials*, Kronmüller H. and Parkin St. (Eds.), John Wiley & Sons (2007)

[19] Metropolis N., Rosenbluth A., Rosenbluth M., Teller A., Teller E., Equation of state calculations by fast computing machines, *J. Chem. Phys.* 21 (1953), 1087-1090

[20] Donckels B., Ghent University, Particle Swarm Optimization, Shuffled Complex Evolution and SIMPSA (Nonlinear Simplex + Simulated Annealing) Matlab toolboxes, available from: http://biomath.ugent.be/~brecht/index.html

[21] Cardoso M. F., Salcedo R. L., Feyo de Azevedo S., The simplex-simulated annealing approach to continuous non-linear optimization, *Comp. Chem. Eng.* 20 (1996), 1065-1080

[22] Duan Q. Y., Gupta V. K., Sorooshian S., Shuffled Complex Evolution Approach for effective and efficient global optimization, *J. Optim. Theory Appl.* 76 (1993), 501-521

[23] Wolpert D. H., Macready W. G., No free lunch theorems for optimization, *IEEE Trans. Evol. Comput.* 1 (1997), 67-82

***Authors**: dr inż. Krzysztof Chwastek, dr inż. Mariusz Najgebauer, prof. nadzw. dr hab. inż. Jan Szczygłowski, Politechnika Częstochowska, Instytut Elektroenergetyki, Al. Armii Krajowej 17, 42-201 Częstochowa, E-mail: krzych / najgebauer / jszczyg@el.pcz.czest.pl.*