

The applications of fixed-point theorem in optimisation problems

KRZYSZTOF CHWASTEK

*Częstochowa University of Technology, Faculty of Electrical Engineering
Armii Krajowej 17, 42-200 Częstochowa, Poland
e-mail: krzych@el.pcz.czyst.pl*

(Received: 04.11.2011, revised: 19.12.2011)

Abstract: The fixed-point theorem is widely used in different engineering applications. The present paper focuses on its applications in optimisation. A Matlab toolbox, which implements the branch-and-bound optimisation method based on the fixed-point theorem, is used for solving different real-life test problems, including estimation of model parameters for the Jiles-Atherton model.

Key words: optimisation, fixed-point theorem, hysteresis, parameter estimation

1. Introduction

The fixed-point theorem introduced by Polish mathematician Stefan Banach (1892-1945) has been widely used in different engineering applications including computational electromagnetism [10], problems of functional analysis e.g. iterative solutions of sets of linear equations, Fredholm and Volterra integral equations, ordinary differential equations [2], as well as in issues related to nonlinear dynamics and chaos theory [18, 22].

The present paper focuses at applications of the theorem for solving chosen optimisation problems. The performance of a Matlab global optimization code, whose driving force is the fixed-point theorem, is checked using some standard optimisation tests. The use of branch-and-bound optimization procedure is shown in greater detail in a practical engineering task – recovery of optimal parameters for a hysteresis model [4].

2. Theoretical foundations

2.1. The Banach contraction theorem

Let us assume that any pair of elements from the set X is assigned a nonnegative real number $\rho(x_1, x_2)$, which fulfills the following propositions (axioms of metric space):

$$\rho(x_1, x_2) = 0 \Leftrightarrow x_1 = x_2 \text{ (non-negativity),} \quad (1)$$

$$\rho(x_1, x_2) = \rho(x_2, x_1) \text{ (symmetry),} \quad (2)$$

$$\rho(x_1, x_2) + \rho(x_2, x_3) \geq \rho(x_1, x_3) \text{ (triangle inequality).} \quad (3)$$

The function $\rho: X \times X \rightarrow \mathfrak{R}_+^1$ with properties (1)-(3) is called a metric or distance on set X , the pair $X = (X, \rho)$ is referred to as metric space. Any subset Y of the metric space X may be considered as a certain (independent) metric space. Its metric is the so-called induced metric, i.e. the metric ρ considered on $Y \times Y$. The space (Y, ρ) is referred to as subspace of the metric space X .

The series $\{x_n\}_{n=1}^{\infty}$ of the components of the metric space X is called a Cauchy series, if $\rho(x_n, x_m) \rightarrow 0$ at $n, m \rightarrow \infty$. If every Cauchy series in a given metric space is convergent, then the metric space is referred to as a complete space.

The mapping $F: X \rightarrow X$ is referred to as a contractive mapping in the metric space X , if there exists a certain constant $\lambda \in (0, 1)$, such that for any pair $x_1, x_2 \in X$ the following inequality holds

$$\rho(Fx_1, Fx_2) \leq \lambda \rho(x_1, x_2). \quad (4)$$

The solution x^* of the equation $x = Fx$ is called a fixed point of the mapping F .

The Banach contraction theorem states that in a complete metric space the contraction mapping has exactly one fixed point x^* . The following equality holds

$$x^* = \lim_{n \rightarrow \infty} x_n, \text{ where } x_0 \in X, x_{n+1} = Fx_n. \quad (5)$$

The estimate of approximation error is given with the inequality

$$\rho(x^*, x_n) \leq \frac{\lambda^n}{1 - \lambda} \rho(x_1, x_0). \quad (6)$$

The proof of the theorem may be found e.g. in [20].

2.2. The continuous global optimisation model

Consider the continuous global optimisation (CGO) model stated as [23]

$$\min f(x) \text{ subject to } x \in D := \{x: l \leq x \leq u \quad g_j(x) \leq 0 \quad j = 1, \dots, n\}. \quad (7)$$

In the above-given model the following notation is used:

$x \in \mathfrak{R}^n$ n -dimensional real-valued vector of decision variables,

$f: \mathfrak{R}^n \rightarrow \mathfrak{R}$ continuous (scalar-valued) objective function,

$D \subset \mathfrak{R}^n$ non-empty set of feasible solutions, a proper subset of \mathfrak{R}^n : this feasible subset is defined by:

$l \in \mathfrak{R}^n, u \in \mathfrak{R}^n$ component-wise finite lower and upper bounds on x , and

$g: \mathfrak{R}^n \rightarrow \mathfrak{R}^m$ a finite collection (m -vector) of continuous constraint functions.

The key analytical assumptions for further examination of the model are:

– l, u are finite vectors; $l \leq x \leq u$ is interpreted component-wise,

– the feasible set $D := \{x : l \leq x \leq u \quad g_j(x) \leq 0 \quad j = 1, \dots, n\}$ is non-empty,
 – f and the components of g are continuous functions. These structural assumptions are sufficient to guarantee the existence of the global solution set X^* ; for any X^* in X^* define $z^* = f(x^*)$. They also support the application of theoretically exact, globally-convergent search methods. In practice, we wish to find numerical estimates of x^* or X^* , and z^* , using efficient global scope search methods.

3. Exemplary test problems

The optimisation problems are common in contemporary engineering practice. A number of test problems with known solutions, useful for checking the performance of optimisation codes, have been developed and discussed in detail elsewhere [1, 7-9, 11, 21, 23].

The first example to be tested is the Rosenbrock valley (or banana) function. It is defined by the formula

$$f(x, y) = (1 - x)^2 + 100(y - x^2)^2. \quad (8)$$

The Rosenbrock function has a global minimum equal to 0 at $(x, y) = (1, 1)$. The shape of Rosenbrock function is depicted in Figure 1.

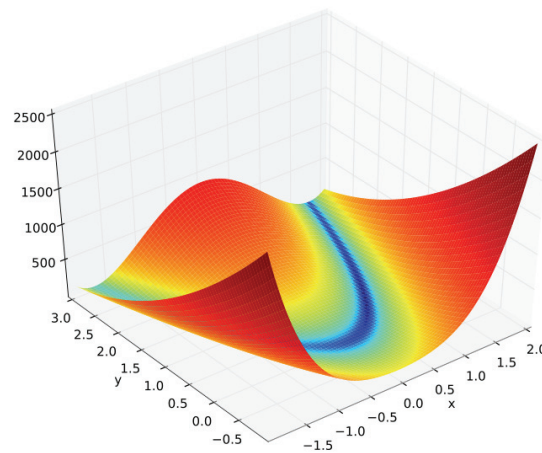


Fig. 1. Rosenbrock function. Source: Wikimedia Commons, public domain

Preset values were 100 iterations, 10 000 evaluations. The algorithm has finished the operation after completing the assumed number of iterations. The solution found was $f(x, y) = 1.5732 \cdot 10^{-9}$ for $(x, y) = (1.0000; 1.0001)$. The optimisation time measured using the “tic; toc” sequence was below 3 seconds on a rather weak AMD Athlon-X2 QL-62 machine running Matlab 7.10.

Another example is the Himmelblau function, defined with the expression

$$f(x, y) = (x^2 + y - 11)^2 + (x + y^2 - 7)^2. \quad (9)$$

The function has one single local maximum at $(x, y) = (-0.270844; -0.923038)$ and four equal local minima equal to zero for the following coordinates: $(x, y) = (3; 2)$; $(x, y) = (-2.805118; 3.131312)$; $(x, y) = (-3.779310; -3.283186)$; $(x, y) = (3.584428; -1.848126)$.

The locations of all the minima can be found analytically, but the expressions are long and complicated. The Himmelblau function is depicted in Figure 2.

It took approximately 48 seconds on the same machine as before (preset values 100 iterations, 10 000 evaluations) to find the approximate minimum at $(x, y) = (-2.8051; 3.1313)$. The algorithm has stopped as the maximum number of fitness function evaluations was exceeded at 70th iteration. The minimum value found was $f(x, y) = 1.5777 \cdot 10^{-9}$.

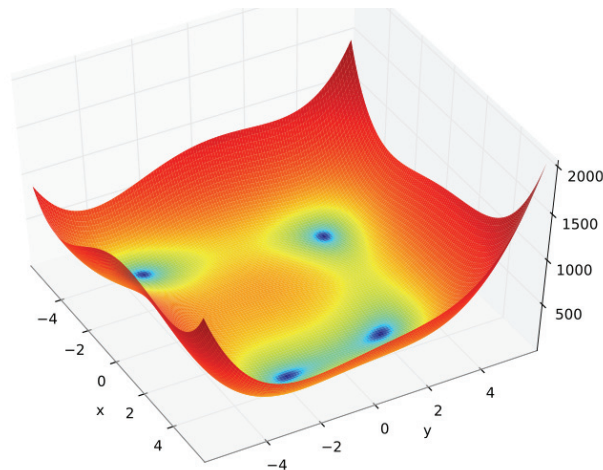


Fig. 2. Himmelblau function. Source: Wikimedia Commons, public domain

The Readers might be curious what caused a substantial increase of computation time in comparison to the one previously examined. The increase was due to a “black-box” operation mode of the algorithm. The toolbox user supplies a definition of fitness function and determines the maximum number of fitness function evaluations and iterations. If either of aforementioned values is exceeded, the algorithm stops. Thus the stopping conditions are set arbitrarily by the user and the algorithm performance may depend on his preferences and experience. In the considered example it was enough to decrease the number of iterations from 100 down to 30 in order to obtain the desired solution within less than 2 seconds. In the second test the final fitness value was $f(x, y) = 1.0293 \cdot 10^{-12}$, which might be treated as a reasonable

approximation to the expected value $f(x, y) = 0$. Figure 3 depicts the variation of fitness function during optimisation process.

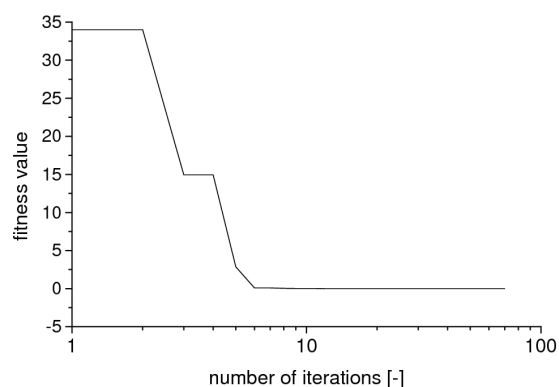


Fig. 3. Variation of fitness value when dealing with Himmelblau function. Source: own work

Yet another example of a test function for global optimisation is the Branin function, defined with the expression

$$f(x, y) = \left(y - \frac{5.1}{4\pi^2} x^2 + \frac{5}{\pi} x - 6 \right)^2 + 10 \left(1 - \frac{1}{8\pi} \right) \cos x + 10. \quad (10)$$

The function is depicted in Figure 4. It has three global minima at $(x, y) = (\pi; 2,275)$; and $(x, y) = (9.42478; 2.475)$. The value of the function at minimum is $f(x, y) = 0.397887$.

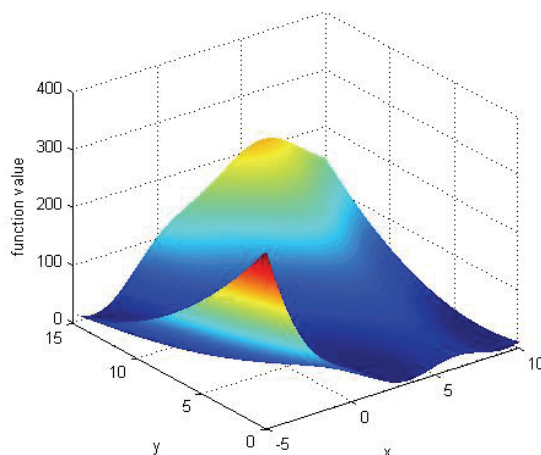


Fig. 4. Branin function. Source: own work

The algorithm was capable to find the global minimum in less than 2.5 seconds on the same machine. Preset values were 100 iterations, 10 000 evaluations.

Throughout the tests, the freeware implementation of DIRECT code, developed by Finkel [8] was used. One should be aware, that there are available some other codes for Matlab and/or other environments, which implement global optimisation routines based on Banach fixed-point theorem, e.g. Multilevel Coordinate Search (MCS) developed by Neumaier [21], commercial TOMLAB [1] or (also commercial) pieces of software developed by Pinter [23]. The choice of the appropriate software tool is up to the potential user. The author has found the Finkel's implementation useful. The algorithm did not require any fine-tuning of working parameters apart from setting the conditions for stopping. MCS seems to be more powerful and flexible than DIRECT, but it requires installation of additional libraries. It should be remarked that DIRECT algorithm performs a transformation of the n-dimensional search space into unit hypercube, thus the obtained solution is equally sensitive to variations of any independent variable. All dimensions of the hypercube are processed simultaneously [4, 8, 16].

4. Estimation of model parameters for the Jiles-Atherton hysteresis model

The Jiles-Atherton (J-A) hysteresis model [14] is a low-dimensional, phenomenological description of magnetization phenomena, based to some extent on physical premises. The model is readily applied by engineers working on coupled phenomena in ferromagnetic cores [3, 12, 13, 17, 19, 24]. One of the drawbacks of the description is the difficult and error-prone method of estimation of optimal model parameters, as proposed by one of model developers [15]. Therefore a number of alternative methods for coping with this task have been developed, cf. e.g. [4, 5, 17]. In the following section, the robust "branch-and-bound" algorithm, based on Banach contraction mapping, shall be applied for this purpose. Its advantage relies in its robustness due to a deterministic mode of operation, what makes it possible to decrease the computation time in comparison to some other approaches, which are stochastic-based (e.g. genetic algorithms, particle swarm).

The so-called inverse J-A model is used. The basic model equation to be integrated numerically is as follows [4, 6, 12, 19, 24]

$$\frac{dM}{dB} = \frac{\delta_M(M_{an} - M) + ck \delta dM_{an} / dH_{eff}}{\mu_0(k\delta + (1 - \alpha)[\delta_M(M_{an} - M) + ck \delta dM_{an} / dH_{eff}])}, \quad (11)$$

where M_{an} is the anhysteretic magnetization, given in the description with the modified Langevin function, $M_{an} = M_s[\coth(H_{eff}/a) - a/H_{eff}]$, δ is the sign of the derivative dB/dt , H_{eff} denotes the so-called effective field, $H_{eff} = H + \alpha M$, whereas a , α , c , k , M_s are the model parameters, whose values are to be determined.

The optimisation problem is thus solved in a five-dimensional space. $\delta_M = 0,5[1 + \text{sign}((M_{an} - M)dB/dt)]$ is introduced in order to avoid negative slopes after a field reversal. For a step-by-step derivation of the relationship (11) the Readers are referred to [24]. The instant values of magnetic field strength H corresponding to the values of magnetisation M are determined from the constitutive relationship $H(t) = B(t)/\mu_0 - M(t)$. The approximate bounds for the model parameters have been determined previously [4]. The fitness value is

given as the squared sum of errors between the measured and the modelled magnetisation values for several data points belonging to the major hysteresis loop. The number of the considered data points is chosen larger than the problem dimension i.e. redundant, so that the effects from inevitable measurement errors could be avoided.

Figure 5 depicts exemplary modelling results for a non-oriented dynamo steel sheet, grade M330-50 A ($500 \times 500 \times 0.5$ mm), excitation frequency approximately 5 Hz. For this frequency it was assumed that dynamic effects due mostly to eddy currents generated in the conductive sheet could be neglected.

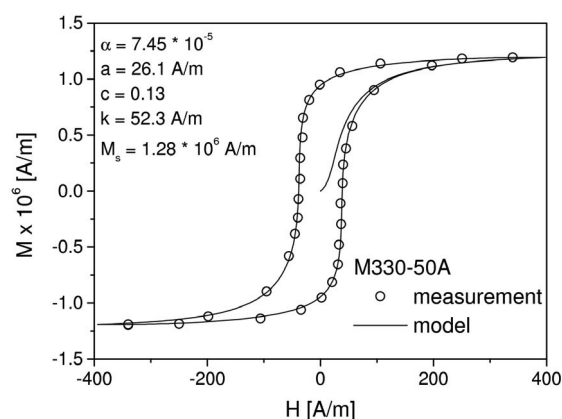


Fig. 5. Quasi-static hysteresis loops for the M330-50 A steel sheet, rolling direction. Source: own work

5. Conclusions

In the papers some chosen applications of the Banach fixed-point theorem in optimisation were presented. At first, the DIRECT algorithm, whose driving force is the fixed-point theorem, was tested on several test problems, used commonly by mathematicians for checking performance of optimization routines. Next, the algorithm was used to recover the optimal parameters for the quasi-static Jiles-Atherton model.

It can be stated that the DIRECT algorithm may be easily applied as a robust and powerful alternative to other global optimisation methods, in particular for low-dimensional problems. The presented Matlab implementation does not require many settings and may be used as a “black-box” even by less experienced users.

Appendix

Listing 1 presents an implementation of the inverse J-A model in the Matlab environment. It is assumed that the input signal is a sine wave, in accordance with international IEC-60404 standards.

```

function dy = jiles (t,y,x)
%y(1) - M, y(2) - H
global Mu0 BMAX F
dy = zeros(2,1);
Heff = (x(1)*y(1)+y(2));
Man = x(5)*(coth(Heff/x(2))-x(2)/Heff);
if abs(Heff)<1E-6 dMan = x(5)/(3*x(2));
else
dMan = x(5)/x(2)*(1 - coth(Heff/x(2))^2 + (x(2)/Heff)^2);
end
deltaM = 1;
if (Man > y(1)) && (cos(2*pi*F*t) < 0)
deltaM = 0; end
if (Man < y(1)) && (cos(2*pi*F*t) > 0)
deltaM = 0; end
licznik = deltaM*(Man - y(1)) +
x(3)*x(4)*signum(cos(2*pi*F*t))*dMan;
dy(1)=1/Mu0*(licznik)/(x(4)* signum(cos(2*pi*F*t)) +
(1-x(1))*licznik)*BMAX*2*pi*F*cos(2*pi*F*t);
dy(2) = BMAX/Mu0*2*pi*F*cos(2*pi*F*t) - dy(1);

```

Listing 2 provides the definition for the fitness function, which is to be minimized:

```

function val = fitme(x, options)
global HREF MREF F BMAX
[t, y] = ode45(@jiles, [0 3.5/(2*F)],
[eps eps], [], x);
indeks_t1 = min(find(t>=2.5/(2*F)));
Htest = y(:,2);
Mtest = y(:,1);
chosenH = Htest(indeks_t1:length(t));
chosenM = Mtest(indeks_t1:length(t));
Mmodel = interp1(chosenH, chosenM, HREF);
val = sum((MREF - Mmodel).^2);

```

Finally, listing 3 shows how to call the branch-and-bound optimisation routine from the Matlab workspace.

```

global F BMAX Mu0 HREF MREF
...
% passing the H and M values in reference points,
% passing the information on excitation amplitude and frequency
...
Hc = interp1(MREF, HREF, 0);
...
% definition of bounds for the search domain
bounds = [0.5*abs(Hc)/max(MREF) 0.7* max(HREF)/max(MREF); 0.5*abs(Hc) 5*abs(Hc); 0 1; 0.5*abs(Hc)
5*abs(Hc); max(MREF) 1.5* max(MREF)];
options.testflag = 0;
options.showits = 1;
options.maxevals = 10000;
options.maxits = 100 ;
Problem.f = 'fitme';
[fmin,xmin,hist] = Direct(Problem,bounds,options);

```


Acknowledgements

The work has been supported by National Center of Science within the framework of research project grant N N510 702540.

References

- [1] Björkmann M., Holmström K., *Global optimization using the DIRECT algorithm in Matlab*. Technical report IMA-TOM-1998-12, Mälardalen University, Västerås, Sweden (1999).
- [2] Bronsztejn I.N., Siemiendajew K.A., Musiol G., Mühlig H., *Nowoczesne kompendium matematyki*. (Taschenbuch der Mathematik – auf Polnisch), PWN Warszawa (2007).
- [3] Chiampi M., Chiarabaglio D., Repetto, M., *A Jiles-Atherton and Fixed-Point combined technique for time periodic magnetic field problems with hysteresis*. IEEE Transactions on Magnetics 31(6): 4306-4311 (1995).
- [4] Chwastek K., Szczygłowski J., *An alternative method to estimate the parameters of Jiles-Atherton model*. Journal of Magnetism and Magnetic Materials 314: 47-51 (2007).
- [5] Chwastek K., Szczygłowski J., *Estimation methods for the Jiles-Atherton model parameters – a review*. Przegląd Elektrotechniczny (Electrotechnical Review) 12: 145-8 (2008).
- [6] Chwastek K., *Modelling of dynamic hysteresis loops using the Jiles-Atherton approach*. Mathematical and Computer Modelling of Dynamical Systems 15(1): 95-105 (2009).
- [7] Dixon L.C.W., Széögö G.P. (Eds.), *Towards Global Optimisation*. vol. 2, North-Holland Publishing Company (1978).
- [8] Finkel D.E., *Global optimization with the DIRECT algorithm*. PhD Thesis, North Carolina State University, Raleigh (2005).
- [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] Hăntilă I.F., *A method for solving stationary magnetic field in nonlinear media*. Revue Roumaine des Sciences Techniques-Électrotechnique et Énergétique 20(3): 397-407 (1975).
- [11] Horst R., Pardalos, P.M. (Eds.), *Handbook of global optimization*. Kluwer (1995).
- [12] Jagieła M., *Calculation of core loss using the Jiles-Atherton model incorporated into magnetic field analysis*. Archives of Electrical Engineering 56(1): 57-69 (2007).
- [13] Jędrzycka C., Sujka P., Szeląg W., *The influence of magnetic hysteresis on magnetorheological fluid clutch operation*. COMPEL 28(3): 711-721 (2009).
- [14] Jiles D.C., Atherton, D.L., *Theory of ferromagnetic hysteresis*. Journal of Magnetism and Magnetic Materials 61(1): 48-60 (1986).
- [15] Jiles D.C., Thøelke J.B., Devine M.K., *Numerical determination of hysteresis model parameters for the modeling of magnetic properties using the theory of ferromagnetic hysteresis*. IEEE Transactions on Magnetics 28(1): 27-35 (1992).
- [16] Jones D.R., Perttunen C.D., Stuckman B.E., *Lipschitzian optimization without the Lipschitz constant*. Journal of Optimization Theory and Applications 79 (1): 157-81 (1993).
- [17] Knypiński Ł., *Optimization and field-circuit simulation of permanent magnet brushless DC motor*. Conference Archives PTETiS 29: 451-456 (2011).
- [18] Kudrewicz J., *Dynamika pętli fazowej*. (Dynamics of phase locked loop – in Polish), WNT Warszawa (1991).
- [19] Łyskawiński W., Sujka P., Szeląg W., Barański, M., *Numerical analysis of hysteresis loss in pulse transformer*. Archives of Electrical Engineering 60(2): 187-192 (2011).
- [20] Muszyński J., Myszkis A.D., *Równania różniczkowe zwyczajne*. (Ordinary differential equations – in Polish), PWN Warszawa (1984).
- [21] Neumaier A., *Complete search in continuous global optimisation and constraint satisfaction*. [In:] Acta Numerica 2004 (A. Iserles, Ed.), Cambridge University Press (2004). See also: <http://www.mat.univie.ac.at/~neum/>.
- [22] Peitgen H.-O., Jürgens H., Saupe, D., *Granice chaosu. Fraktale*. cz. 1 (Polish translation of *Chaos and fractals. New frontiers of science*.), PWN Warszawa (1995).

-
- [23] Pintér J., *Nonlinear optimization with GAMS/LGO*. Journal of Global Optimization 38: 79-101 (2007). See also: <http://www.pinterconsulting.com>.
- [24] Sadowski N., Batistela N.J., Bastos J.P.A., Lajoie-Mazenc M., *An inverse Jiles-Atherton model to take into account hysteresis in time-stepping finite-element calculations*. IEEE Transactions on Magnetism 38(2): 797-800 (2002).