Proceedings of the International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2008
13–17 June 2008.

Hybrid Optimization Coupling Electromagnetism and Descent Search for Engineering Problems

Ana Maria A.C. Rocha[1] and Edite M.G.P. Fernandes[1]

[1] *Production and Systems Department, University of Minho, 4710-057 Braga, Portugal*

emails: `arocha@dps.uminho.pt`, `emgpf@dps.uminho.pt`

Abstract

In this paper, we present a new stochastic hybrid technique for constrained global optimization. It is a combination of the electromagnetism-like (EM) mechanism with an approximate descent search, which is a derivative-free procedure with high ability of producing a descent direction. Since the original EM algorithm is specifically designed for solving bound constrained problems, the approach herein adopted for handling the constraints of the problem relies on a simple heuristic denoted by feasibility and dominance rules. The hybrid EM method is tested on four well-known engineering design problems and the numerical results demonstrate the effectiveness of the proposed approach.

Key words: Hybrid methods, electromagnetism-like mechanism, descent search
MSC 2000: 90C15, 90C56, 90C30

1 Introduction

The problem that is addressed in the paper considers finding a global solution of a nonlinear optimization problem in the following form:

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & g(x) \leq 0, \; x \in \Omega, \end{array} \tag{1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ and $g : \mathbb{R}^n \to \mathbb{R}^p$ are nonlinear continuous functions and $\Omega = \{x \in \mathbb{R}^n : l \leq x \leq u\}$ is a closed set. We assume that the objective function f is nonconvex and possesses many local minima in the feasible region. This class of global optimization problems is very important and frequently encountered in engineering applications. The most common approach for solving constrained optimization problems is based on penalty functions. Penalty terms are added to the objective function to penalize constraints violation. The penalty techniques transform the constrained problem into an unconstrained problem by penalizing f when constraints are violated and then minimizing the penalty function using methods for unconstrained problems [1, 4, 12, 14].

The approach herein adopted for handling the constraints $g(x) \leq 0$ of problem (1) relies on a simple heuristic consisting on three selective rules, denoted by feasibility and dominance (FAD) rules [10, 17]. Other techniques that aim to preserve feasibility are adopted in [8, 9, 15, 16].

In this paper, we are interested in the electromagnetism-like (EM) algorithm proposed in [2]. This is a population-based algorithm that simulates the electromagnetism theory of physics by considering each point in the population as an electrical charge. The method uses an attraction-repulsion mechanism to move a population of points towards optimality. The EM algorithm is specifically designed for solving optimization problems with bound constraints [1, 2, 3]. Like other hybrid population set-based algorithms, the original EM algorithm incorporates a local search algorithm to exploit the local minima around the best point of the population. In this paper, we will use a derivative-free heuristic method to produce an approximate descent direction for the objective function, at the best point of the population, followed by a backtracking line search.

The paper is organized as follows. In Section 2 we describe the EM algorithm that incorporates the FAD rules for constraint-handling. Section 3 presents the hybridization of the EM algorithm with the descent search heuristic and Section 4 contains the results of the numerical experiments on four benchmark engineering optimization problems. We conclude the paper in Section 5.

2 The electromagnetism optimization algorithm

In this section we describe the EM algorithm for solving problem (1). The algorithm starts with a population of randomly generated points from the feasible set Ω. Analogous to electromagnetism, each point is a charged particle that is released to the space. The charge of each point is related to the objective function value and determines the magnitude of attraction of the point over the others in the population. The better the objective function value, the higher the magnitude of attraction. The charges are used to find a direction for each point to move in subsequent iterations. The regions that have higher attraction will signal other points to move towards them. In addition, a repulsion mechanism is also introduced to explore new regions for even better solutions [2, 3]. The EM algorithm is presented below and comprises four main procedures.

Algorithm 1 (Electromagnetism)
input: Nit_{max}, $LsIt_{max}$ δ;
$k \leftarrow 0$
Initialize()
while $k \leq Nit_{max}$ ***do***
  $F \leftarrow$ *CalcF()*
  Move(F)
  Local($LsIt_{max}$, δ*)*
  $k \leftarrow k + 1$
end while

A more detailed explanation of the EM algorithm for solving constrained problems follows. In the *Initialize* procedure, a population of p_{size} points is randomly generated from the feasible region Ω. Let x^i be the ith point of the population. Then each coordinate of a point, denoted as x^i_k ($k = 1, \ldots, n$), is computed by

$$x^i_k = l_k + \lambda(u_k - l_k) \tag{2}$$

where $\lambda \sim U(0, 1)$. Then, all points are evaluated and compared in order to identify the best point, x^{best}.

The comparison between two points to the selection of the most promising point is made according to the so called FAD rules, as follows,

(1) among two feasible points, the one that has better objective function value is preferred;

(2) any feasible point is preferred to any infeasible solution;

(3) among two infeasible points, the one that has smaller constraints violation is preferred.

The constraints violation is measured by

$$\mathrm{CViol}(x) = \left(\sum_{j=1}^{p} \left(\max \{0, g_j(x)\} \right)^2 \right)^{1/2}.$$

Hence, a point x^i with $\mathrm{CViol}(x^i) = 0$ is feasible, whereas the point is infeasible if $\mathrm{CViol}(x^i) > 0$. The reader is referred to [13] for more details.

For the *CalcF* procedure, the Coulomb's law of the electromagnetism theory is used. It states that the force exerted on a point via other points is inversely proportional to the square of the distance between the points and directly proportional to the product of their charges. In each iteration, we compute the charges of the points according to their objective function values. As the charge q^i of point x^i determines the power of attraction or repulsion for that point, the charge is computed according to the objective function value by

$$q^i = \exp \left(\frac{-n|f(x^i) - f(x^{best})|}{\sum_{j=1}^{p_{size}} |f(x^j) - f(x^{best})|} \right) \quad \text{for} \quad i = 1, \ldots, p_{size}. \tag{3}$$

In this way the points that have better objective function values possess higher charges. This is a scaled distance of the function value at x^i to the function value of the best point in the population.

Since the charges (3) are all positive, the direction of a force F^i_j depends on the comparison of the points x^i and x^j, according to the FAD rules. Hence, the statement x^j is better than x^i means that x^j is the preferred point according to the FAD rules and the point x^j attracts x^i and consequently the direction of the force should be $\overrightarrow{x^i x^j}$.

Otherwise, if x^i is the preferred point according to the FAD rules then x^j repels x^i and the direction of the force is $\overrightarrow{x^j x^i}$. Thus,

$$F_j^i = \begin{cases} (x^j - x^i)\frac{q^i q^j}{\|x^j - x^i\|^3} & \text{if } x^j \text{ is better than } x^i \\ \\ (x^i - x^j)\frac{q^i q^j}{\|x^j - x^i\|^3} & \text{otherwise,} \end{cases} \tag{4}$$

for $j \neq i$. Then the total force vector F^i exerted on each point x^i by the other $p_{size} - 1$ points is calculated by adding the individual component forces, F_j^i,

$$F^i = \sum_{j \neq i}^{p_{size}} F_j^i, \quad i = 1, \ldots, p_{size}.$$

In the *Move* procedure, the total force vector, F^i, is used to move the point x^i in the direction of the force by a random step length $\lambda \sim U(0,1)$ as follows

$$x_k^i = \begin{cases} x_k^i + \lambda \frac{F_k^i}{\|F^i\|}(u_k - x_k^i) & \text{if } F_k^i > 0 \\ \\ x_k^i + \lambda \frac{F_k^i}{\|F^i\|}(x_k^i - l_k) & \text{otherwise} \end{cases} \tag{5}$$

for each coordinate k ($k = 1, 2, \ldots, n$) and for $i = 1, \ldots, p_{size}$ and $i \neq best$. Notice that the best point, x^{best}, is not moved and is carried out to the subsequent iteration.

The *Local* procedure performs a local refinement and is applied to one point in the population. The local search presented in [2] is a random line search algorithm that is applied coordinate by coordinate only to the best point in the population. First, based on the parameter δ, the procedure computes the maximum feasible step length,

$$s_{max} = \delta(\max_{1 \leq k \leq n}(u_k - l_k)).$$

Then, for each coordinate k ($k = 1, 2, \ldots, n$), a random number λ between 0 and 1 is selected as a step length and a new point y is componentwise calculated along that direction by

$$y_k = x_k^{best} + \lambda\, s_{max}.$$

If an improvement is observed, according to the FAD rules, within $LsIt_{max}$ iterations, the best point is replaced by y and the search along that coordinate k ends.

3 Electromagnetism-like hybridization

In this section, a hybridization of the electromagnetism-like algorithm with a descent search is described. The basic idea behind this hybridization is the combination of global techniques from the EM method with local techniques of a derivative-free heuristic method that produces an approximate descent direction and consequently generates a new promising point. Thus, instead of using the *Local* search procedure described in Section 2 we apply a descent local search procedure to refine the best point.

This local search procedure is an iterative stochastic optimization method that generates a sequence of approximations of the optimizer by assuming an approximate descent direction. It relies on:

(1) a set of two exploring points, randomly generated from the neighborhood of the best point;

(2) an approximate descent direction for f at x^{best};

(3) a generation of a new promising point along the unit length descent direction with a prescribed scalar step size;

(4) the selection of the most promising point according to the so called FAD rules.

In the following subsections, a brief presentation of the procedures involved in the hybridization of the electromagnetism-like algorithm with a descent search is made. The local search algorithm in the hybrid EM algorithm is then presented.

3.1 Generating exploring points

Here, we introduce the algorithm that aims to randomly generate two random exploring points x_i^{rand}, $i = 1, 2$, in the neighborhood of the best point x^{best}. Each x_i^{rand} is generated so that

$$\|x^{best} - x_i^{rand}\| \leq \varepsilon_r \qquad \text{for} \quad i = 1, 2,$$

for a sufficiently small positive value of ε_r, as described in the following algorithm:

Algorithm 2 (Generate)
input: ε_r, x^{best}
for $k = 1, \ldots, n$ ***do***
    for $i = 1, 2$ ***do***
        $\lambda_1 \leftarrow U(0, 1)$
        $\lambda_2 \leftarrow U(0, 1)$
        if $\lambda_1 > 0.5$ ***then***
            $x_{i,k}^{rand} \leftarrow x_k^{best} + \lambda_2 \varepsilon_r$
        else
            $x_{i,k}^{rand} \leftarrow x_k^{best} - \lambda_2 \varepsilon_r$
        end if
    end for
end for

3.2 An approximate descent search

Here, we describe a strategy to generate an approximate descent direction, d, for the objective function f, at the best point x^{best}. Based on the two random exploring points previously described, a descent direction is generated by

$$d = -\frac{1}{\sum_{j=1}^{2} |\Delta f_j|} \sum_{i=1}^{2} \Delta f_i \frac{x^{best} - x_i^{rand}}{\|x^{best} - x_i^{rand}\|}, \qquad (6)$$

where $\Delta f_j = f(x^{best}) - f(x_j^{rand})$.

Theoretical properties related to this direction vector are shown in [7].

3.3 Generate new point

The descent direction, d, is used to move the best point in that direction, as can be seen in Algorithm 3, where $0 < \alpha \leq 1$ represents the step size. We remark that bound constraints feasibility is maintained by using the normalized descent direction and scaling it with the allowed range of movement towards the lower bound l_k, or the upper bound u_k, of the set Ω.

Algorithm 3 (New point)
input: α, d, u, l, x^{best}
for $k = 1, \ldots, n$ ***do***
  if $d_k > 0$ ***then***
    $y_k \leftarrow x_k^{best} + \alpha \frac{d_k}{\|d_k\|}(u_k - x_k^{best})$
  else
    $y_k \leftarrow x_k^{best} + \alpha \frac{d_k}{\|d_k\|}(x_k^{best} - l_k)$
  end if
end for

3.4 The hybrid EM algorithm

We now present a more detailed and formal description of the local search in the electromagnetism-like mechanism with the descent search hybridization (Algorithm 4).

In this hybrid EM algorithm we implement a backtracking line search to progress towards optimality, as follows. First, we generate two exploring points and a descent direction. These two steps in the Algorithm 4 are executed whenever *flag* is set to 1. Then, a new point y is calculated and, according to the FAD rules, either y or x^{best} is preferred. If x^{best} is the preferred point, then y is discarded, the step size is halved (*i.e.*, $\alpha \leftarrow \alpha/2$) and a new point is evaluated along the same descent direction (*flag* is set to 0 in the Algorithm 4). However, when y is preferred, another approximate descent direction for f, at y, is computed (*flag* is set to 1 and the step size, α, is re-initialized to 1) and the process is repeated.

Algorithm 4 below describes the steps of the local search procedure in the hybrid EM method. The setting of parameters used in this algorithm is discussed later on in Section 4.

Algorithm 4 (Local search in the hybrid EM)
input: $LsIt_{max}$, x^{best}
$flag \leftarrow 1$, $\alpha \leftarrow 1$, $k \leftarrow 0$
while $k \leq LsIt_{max}$ ***do***
  if $flag = 1$ ***then***
    Generate(x^{rand})
    Compute descent direction d *using (6)*
  end if
  New point(y)
  if *both* y *and* x^{best} *are feasible* ***then***
    if $f(y) < f(x^{best})$ ***then***
      $x^{best} \leftarrow y$, $\alpha \leftarrow 1$, $flag \leftarrow 1$
    else
      $\alpha \leftarrow \alpha/2$, $flag \leftarrow 0$
    end if
  end if
  if y *is feasible and* x^{best} *is infeasible* ***then***
    $x^{best} \leftarrow y$, $\alpha \leftarrow 1$, $flag \leftarrow 1$
  end if
  if y *is infeasible and* x^{best} *is feasible* ***then***
    $\alpha \leftarrow \alpha/2$, $flag \leftarrow 0$
  end if
  if *both* y *and* x^{best} *are infeasible* ***then***
    if $CViol(y) > CViol(x^{best})$ ***then***
      $\alpha \leftarrow \alpha/2$, $flag \leftarrow 0$
    else
      $x^{best} \leftarrow y$, $\alpha \leftarrow 1$, $flag \leftarrow 1$
    end if
  end if
  $k \leftarrow k + 1$
end while

4 Numerical experiments

Problems of practical interest are important for assessing the effectiveness of a given approach. Thus, to evaluate the performance of the herein proposed hybrid electromagnetism-like algorithm for constrained problems, a set of 4 benchmark engineering problems, described in full detail in [11] (see also [9] and [16]) is used. A comparison with other published results is also included.

The algorithm is coded in the C programming language and it contains an interface to connect to AMPL so that the problems coded in AMPL could be easily solved [6].

AMPL is a mathematical programming language that allows the codification of optimization problems in a powerful and easy to learn language. The set of coded problems may be obtained from the first author upon request.

We tested two versions of the EM algorithm both incorporating the FAD rules for constraint-handling: the EM with the original local search procedure, as described in Section 2, denoted in the subsequent tables only by EM; and the hybrid EM with the local descent search as presented in Section 3. For all problems, the used parameters are as follows: $\delta = 0.001$, $LsIt_{max} = 10$, $Nit_{max} = 5000$, $\varepsilon_r = 0.001$. As required by any stochastic algorithm, we record values of the best function value, f_{best}, the average best function values, f_{avg}, the standard deviation, SD, and the worst function value, f_{worst}, obtained after 100 independent runs, each starting from a random population with different seeds. Here, we consider a population size of 20 points and use a limit of Nit_{max} iterations to terminate the algorithms.

4.1 The problems set

We now summarize the characteristics of the chosen engineering problems:

- **Design of a welded beam** [9, 11]. In this problem we minimize the cost of a welded beam, subject to constraints on the shear stress, bending stress in the beam, buckling load on the bar, end deflection of the beam, and side constraints. There are 4 design variables (h, l, t and b) and 7 inequality constraints;
- **Design of a tension/compression spring** [9, 11]. This problem minimizes the weight of a tension/compression spring, subject to constraints on the minimum deflection, shear stress, surge frequency, limits on outside diameter and on design variables. The problem has 3 design variables (d, D and N) and 4 inequality constraints;
- **Design of a gear train** [11]. This problem minimizes the cost of a gear ratio of a gear train, subject to constraints on the design variables. The problem has 4 design variables (n_A, n_B, n_C and n_D) constrained in $[12, 60]$;
- **Design of a pressure vessel** [9, 11]. This problem consists of minimizing the total cost of the material, forming and welding of a cylindrical pressure vessel. The problem has 4 design variables (T_s, T_h, R and L) and 4 inequality constraints.

4.2 Comparative results

We begin by reporting the results obtained when solving the welded beam design problem. Table 1 contains the best, average and worst results obtained by the EM and hybrid EM algorithms. For comparison, the best results reported in [11] are also included. The population based algorithm therein proposed is a standard Unified Particle Swarm Optimization (UPSO) that has been implemented for four values of a particular parameter u: 0, 0.2, 0.5 and 1. This is a combination of the standard global and local PSO versions. The EM algorithm found a solution having objective function

Table 1: Best, average and worst results for the welded beam design problem.

Method	f_{best}	f_{avg}	SD	f_{worst}
UPSO($u = 0.5$) [11]	1.76558	1.96820	1.55415e-1	2.84406
EM	1.726785	1.776614	3.045007e-2	1.862183
hybrid EM	1.725311	1.750363	1.965722e-2	1.802470

value within 0.1% of the best-known solution, in 115308 function evaluations, while the hybrid EM found a solution within 0.03% of the best-known solution, in 155082 function evaluations. Table 2 contains the values of the design variables, as well as the values of the constraints at the best solution found, over the 100 runs. For comparative purposes, we include similar results published in [5], which implements a genetic-based algorithm, and in [9] and [16], which are PSO type methods.

Table 2: Comparative results for the beam problem.

Values	Best solution found				
	EM	hybrid EM	in [5]	in [9]	in [16]
h	0.204359	0.205651	0.2088	0.20573	0.201381
l	3.500343	3.473614	3.4205	3.47049	3.23192
t	9.036422	9.036222	8.9975	9.03662	10.0
b	0.205740	0.205759	0.2100	0.20573	0.201381
g_1	-6.534397e-2	-4.009198	3.37812e-1	0.0	-2.92784e-2
g_2	-1.183849e-1	-1.679412	-353.902604	0.0	-4972.77
g_3	-1.381052e-3	-1.081644e-4	-1.20e-3	-5.551115e-11	-3.58641e-7
g_4	-3.430331	-3.432551	-3.411865	-3.432984	-3.32625
g_5	-7.935900e-2	-8.065100e-2	-8.380e-2	-8.07296e-2	-7.63803e-2
g_6	-2.355401e-1	-2.355405e-1	-2.35649e-1	-2.355403e-1	-2.39099e-1
g_7	-7.888930e-1	-2.432333	-363.232384	-9.094947e-13	-3.91764e-3
f	**1.726785**	**1.725311**	**1.74830941**	**1.72485084**	**1.81429**

Tables 3 and 4 contain the results of the spring design problem. The best solution found by EM is 0.003% below the best-known solution, after 109769 function evaluations, while the hybrid EM found a solution 0.006% below the best-known solution, using 155076 objective function evaluations.

Table 5 contains the f_{best}, f_{avg} and f_{worst} obtained after 100 runs for the UPSO method of [11], as well as for the EM and hybrid EM algorithms, when solving the gear train design problem. EM found a solution in 115069 function evaluations, while the hybrid EM needed 155033 objective function evaluations. Table 6 presents detailed results concerning the design variables of the problem.

Table 3: Best, average and worst results for the tension/compression spring design problem.

Method	f_{best}	f_{avg}	SD	f_{worst}
UPSO($u = 0.5$) [11]	1.28158e-2	4.67351e-2	2.14505e-1	1.57998
EM	1.266581e-2	1.283445e-2	1.997469e-4	1.380261e-2
hybrid EM	1.266535e-2	1.269072e-2	3.039597e-5	1.279914e-2

Table 4: Comparative results for the spring problem.

Values	Best solution found				
	EM	hybrid EM	in [5]	in [9]	in [16]
d	0.051610	0.051755	0.051480	0.051466369	0.05
D	0.354808	0.358310	0.351661	0.351383949	0.310414
N	11.402126	11.196240	11.632201	11.60865920	15.0
g_1	-3.941677e-6	-3.553156e-7	-2.080e-3	-3.336613e-3	-3.30997e-6
g_2	-1.782013e-5	-1.234269e-6	-1.10e-4	-1.0970128e-4	-1.73742e-2
g_3	-4.049882	-4.056911	-4.026318	-4.0263180998	-186.267
g_4	-7.290545e-1	-7.266231e-1	-7.31239e-1	-7.312393333e-1	-7.59724e-1
f	**0.01266581**	**0.01266535**	**0.0127047834**	**0.0126661409**	**0.0131926**

Table 5: Best, average and worst results for the gear train design problem.

Method	f_{best}	f_{avg}	SD	f_{worst}
UPSO($u = 0$) [11]	2.70085e-12	3.92135e-8	7.71670e-8	6.41703e-7
EM	1.307208e-21	2.136513e-16	3.884946e-16	3.192848e-15
hybrid EM	2.040657e-18	8.998912e-15	1.855748e-14	1.093523e-13

Table 6: Comparative results for the train problem.

Method	n_A	n_B	n_C	n_D	f
EM	54.208242	15.842449	15.825780	32.056642	**1.307208e-21**
hybrid EM	55.036626	17.362821	26.994790	59.026132	**2.040657e-18**

Table 7: Best, average and worst results for the pressure vessel design problem.

Method	f_{best}	f_{avg}	SD	f_{worst}
UPSO($u = 0.5$) [11]	6.15470e3	8.01637e3	7.45869e2	9.38777e3
EM	5.894836e3	6.383338e3	4.766863e2	7.320809e3
hybrid EM	5.911713e3	6.462390e3	5.665410e2	8.944713e3

Table 8: Comparative results for the vessel problem.

Values	Best solution found				
	EM	hybrid EM	in [5]	in [9]	in [16]
T_s	0.783512	0.780817	0.8125	0.812	0.778169
T_h	0.387376	0.393103	0.4375	0.4375	0.384649
R	40.596075	40.45676	40.3239	42.09845	40.3196
L	196.186997	198.147519	200.0	176.6366	200.0
g_1	-7.866810e-6	-1.593979e-6	-3.4324e-2	0.0	-9.94553e-9
g_2	-8.984847e-5	-7.145636e-3	-5.2847e-2	-3.588e-2	-3.80778e-9
g_3	-3.943287e-2	-246.0737	-27.105845	-5.8208e-11	-5.84856e-4
g_4	-43.813003	-41.852481	-40.0000	-63.3634	-40.0
f	**5894.835806**	**5911.713232**	**6288.7445**	**6059.131296**	**5885.33**

Finally, when solving the pressure vessel design problem, we obtain the results reported in Tables 7 and 8. The best results obtained by the EM and hybrid EM algorithms are better than most of the until now published. The solution obtained by EM is 5894.835806 (with 115310 function evaluations). The hybrid EM found the solution 5911.713232 after 155079 function evaluations.

5 Conclusions

This paper presents a new version of the electromagnetism-like optimization algorithm for solving global constrained optimization problems. This version relies on simple rules to maintain feasibility instead of implementing a penalty technique [1], avoiding therefore the update of the penalty parameter that is associated with the penalization of the constraints in the penalty function. Further, we hybridize the electromagnetism-like mechanism and a local descent search in order to get a better movement and to refine the best point in the population. A basic backtracking line search technique is also included to give faster progress towards optimality.

To assess the performance of the proposed hybrid EM algorithm, a set of four constrained engineering problems of practical interest is solved. A comparison with other stochastic-type methods is included. The results show the effectiveness of our hybrid EM method. Embedding a fitness function that does not need any penalty parameter in the charge calculation of the EM algorithm is a matter for future research.

References

[1] S. I. Birbil, *Stochastic Global Optimization techniques*, PhD thesis, North Carolina State University, 2002.

[2] S. I. Birbil and S.-C. Fang, *An electromagnetism-like mechanism for global optimization*, Journal of Global Optimization, **25** (2003) 263–282.

[3] S. I. Birbil, S.-C. Fang, and R.-L. Sheu, *On the convergence of a population-based global optimization algorithm*, Journal of Global Optimization, **30** (2004) 301–318.

[4] E. G. Birgin, C. A. Floudas, and J. M. Martínez, *Global minimization using an augmented Lagrangian method with variable lower-level constraints*, Report, November 2006.

[5] C.A. Coello Coello, *Use of a self-adaptive penalty approach for engineering optimization problems*, Computers in Industry, **41** (2000) 113–127.

[6] R. Fourer, D. M. Gay and B. Kernighan, *A modeling language for mathematical programing*, Management Science, **36** (1990) 519–554 (http://www.ampl.com).

[7] A.-R. Hedar and M. Fukushima, *Heuristic pattern search and its hybridization with simulated annealing for nonlinear global optimization*, Optimization Methods and Software, **19** (2004) 291–308.

[8] X. Hu and R. Eberhart, *Solving constrained nonlinear optimization problems with particle swarm optimization.* Proceedings of 6th World Multiconference on Systemics, Cybernetics and Informatics, 2002.

[9] X. Hu, R. C. Eberhart, and Y. Shi, *Engineering optimization with particle swarm*, Proceedings of 2003 IEEE Swarm Intelligence Symposium, 53–57, 2003.

[10] D. Karaboga and B. Basturk, *Artificial bee colony (ABC) optimization algorithm for solving constrained optimization problems*, Vol 4529 of LNAI, 789–798. Springer-Verlag, IFSA 2007 edition, 2007.

[11] K. E. Parsopoulos and M. N. Vrahatis, *Unified particle swarm optimization for solving constrained engineering optimization problems*, Vol 3612 of LNCS, 582–591. Springer-Verlag, ICNC 2005 edition, 2005.

[12] Y. G. Petalas, K. E. Parsopoulos and M. N. Vrahatis, *Memetic particle swarm optimization*, Annals of Operations Research, **156** (2007) 99–127.

[13] A. M. A. C. Rocha and E. M. G. P. Fernandes, *Feasibility and dominance rules in the electromagnetism-like algorithm for constrained global optimization problems*, Computational Science and Applications, Vol 5073 of LNCS (in press) Springer-Verlag, July 2008.

[14] M.-J. Tahk, H.-W. Woo, and M.-S. Park, *A hybrid optimization method of evolutionary and gradient search*, Engineering Optimization, **39** (2007) 87–104.

[15] G. Toscano and C. C. Coello, *A constraint-handling mechanism for particle swarm optimization*, Proceedings of the 2004 Congress on Evolutionary Computation, 1396–1403, 2004.

[16] A. I. F. Vaz and E. M. G. P. Fernandes, *Optimization of nonlinear constrained particle swarm*, Technological and Economic Development of Economy, **12** (2006) 30–36.

[17] A. E. M. Zavala, A. H. Aguirre, and E. R. V. Diharce, *Constrained optimization via particle evolutionary swarm optimization algorithm (PESO), In GECCO'05* (2005) 209–216.