Chapter 12

A modified electromagnetism-like algorithm based on a pattern search method

Ana Maria A. C. Rocha, Edite M. G. P. Fernandes

Minho University, Campus de Gualtar, 4710-057 Braga, Portugal
{arocha;emgpf}@dps.uminho.pt

Abstract. The Electromagnetism-like (EM) algorithm, developed by Birbil and Fang [2] is a population-based stochastic global optimization algorithm that uses an attraction-repulsion mechanism to move sample points towards optimality. A typical EM algorithm for solving continuous bound constrained optimization problems performs a local search in order to gather information for a point, in the population. Here, we propose a new local search procedure based on the original pattern search method of Hooke and Jeeves, which is simple to implement and does not require any derivative information. The proposed method is applied to different test problems from the literature and compared with the original EM algorithm.

Keywords. Global optimization, electromagnetism-like algorithm, pattern search method.

12.1 Introduction

Many real life global optimization problems that arise in areas such as physics, chemistry, and molecular biology, involve multi-modal and non-differentiable nonlinear functions of many variables that are difficult to handle by conventional gradient-based algorithms. As a result, many researchers have devoted themselves in finding reliable stochastic global optimization methods that do not require any derivative computation. Recent-

ly, Birbil and Fang proposed the electromagnetism-like (EM) algorithm that is a population-based stochastic search method for global optimization [1,2]. This algorithm 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 original algorithm incorporates a simple random local search procedure that is applied coordinate by coordinate to one point only or to all points in the population [2]. In an attempt to improve the accuracy of the results and to accelerate convergence we propose a modification to the original EM algorithm by replacing the random local search with a pattern search method [6] with guaranteed convergence.

The method is to work on nonlinear optimization problems with box constraints in the following form:

$$\min f(x), \quad \text{subject to } x \in \Omega \tag{12.1}$$

where $f : \Re^n \to \Re$ is a nonlinear function and $\Omega = \{x \in \Re^n : l \le x \le u\}$ is a bounded feasible region.

The paper is organized as follows. Sect. 12.2 briefly introduces the original EM algorithm and Sect. 12.3 is devoted to describe the main ideas concerning the Hooke and Jeeves pattern search method. Sect. 12.4 contains the numerical results and some conclusions are drawn in Sect. 12.5.

12.2 Electromagnetism-like algorithm

The EM algorithm starts with a population of randomly generated points from the feasible region. 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 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. The following notation is used: $x^i \in \Re^n$ denotes the ith point of a population; x^{best} is the point that has the least objective function value; $x_k^i \in \Re$ ($k = 1,...,n$), is the kth coordinate of the point x^i of the population; m is the number of points in the population; MaxIt is the maximum number of EM iterations; LSIt denotes the maxi-

mum number of local search iterations; and δ is a local search parameter, $\delta \in [0,1]$.

The EM algorithm comprises four main procedures.

Algorithm EM(m, MaxIt, LSIt, δ)
Initialize()
iteration $\leftarrow$ 1
while termination criteria are not satisfied **do**
  Local(LSIt,δ)
  $F \leftarrow$ *CalcF*()
  Move(F)
  iteration $\leftarrow$ iteration + 1
end while

Details of each procedure follow. *Initialize* is a procedure that aims to randomly generate m points from the feasible region. Each coordinate of a point x_k^i ($k = 1,...,n$) is assumed to be uniformly distributed between the corresponding upper and lower bounds, i.e., $x_k^i = l_k + \lambda\ (u_k - l_k)$ where $\lambda \sim U(0,1)$. After computing the objective function value for all the points in the population, the procedure identifies the best point, x^{best}, which is the point with the best function value.

The *Local* procedure performs a local refinement and can be applied to one point or to all points 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, the procedure computes the maximum feasible step length, $Length = \delta\,(max_k\,(u_k - l_k))$, based on δ. This quantity is used to guarantee that the local search generates feasible points. Second, the best point is assigned to a temporary point y to store the initial information. Next, for each coordinate k, a random number λ between 0 and 1 is selected as a step length and the point y_k is moved along that direction, $y_k = y_k + \lambda\ Length$. If an improvement is observed, within LSIt iterations, the best point is replaced by y and the search along that coordinate ends.

The *CalcF* procedure aims to compute the total force exerted on a point via other points. First a charge-like value, q^i, that determines the power of attraction or repulsion for the point x^i, is assigned. The charge of the point is calculated according to the relative efficiency of the objective function values, i.e.,

$$q^i = \exp\left(-n\frac{(f(x^i) - f_{best})}{\sum_{j=1}^{m}(f(x^j) - f_{best})}\right),\ i = 1,...,m \tag{12.2}$$

Hence, the points that have better objective function values possess higher charges. The total force vector F^i exerted on each point is calculated by adding the individual component forces, F^i_j, between any pair of points x^i and x^j,

$$F^i = \sum_{j \neq i}^{m} F^i_j = \begin{cases} (x^j - x^i) q^i q^j / (\| x^j - x^i \|^3), & \text{if } f(x^j) < f(x^i) \\ (x^i - x^j) q^i q^j / (\| x^j - x^i \|^3), & \text{if } f(x^j) \geq f(x^i) \end{cases} \tag{12.3}$$

$i = 1,...,m$.

Finally, the *Move* procedure uses the total force vector F^i, to move the point x^i in the direction of the force by a random step length,

$$x^i_k = x^i_k + \lambda \frac{F^i}{\| F^i \|} (RNG), i = 1,2,\ldots,m \text{ and } i \neq best \tag{12.4}$$

where *RNG* is a vector that contains the allowed range of movement towards the lower bound l_k, or the upper bound u_k, for each coordinate k. The random step length λ is assumed to be uniformly distributed between 0 and 1. Note that feasibility is maintained by using the normalized force exerted on each point. The best point, x^{best}, is not moved and is carried out to the subsequent iteration.

12.3 Hooke and Jeeves pattern search method

In this section, we describe our modification to the original EM algorithm. In this algorithm, the *Local* procedure is based on a random line search method [2]. Here a new *Local* procedure based on the Hooke and Jeeves (HJ) pattern search algorithm is proposed. This is a derivative-free method that searches in the neighbourhood of a point x^i for a better approximation via exploratory and pattern moves [4,6]. To reduce the number of function evaluations, the HJ pattern search algorithm is applied to the current best point only. This algorithm is a variant of the coordinate search, in the sense that incorporates a pattern move to accelerate the progress of the algorithm, by exploiting information obtained from the search in previous successful iterations. The exploratory move carries out a coordinate search (a search along the coordinate axes) about the best point, with a step length δ. If a new trial point, y, with a better function value than x^{best} is encountered, the iteration is successful. Otherwise, the iteration is unsuccessful and δ should be reduced. If the previous iteration was successful, the vector $y - x^{best}$ defines a promising direction and a pattern move is then implemented, meaning that the exploratory move is carried out about the trial

point $y + (y - x^{best})$, rather than about the current point y. Then, if the coordinate search is successful, the returned point is accepted as the new point; otherwise, the pattern move is rejected and the method reduces to coordinate search about y. Please see [4] for details. To ensure feasibility in the HJ pattern search algorithm an exact penalty strategy is used. This technique considers solving

$$\min F(x) \equiv \begin{cases} f(x) & \text{if } x \in \Omega \\ \infty & \text{otherwise} \end{cases} \tag{12.5}$$

rather than problem (12.1). This means that any infeasible trial point is rejected, since the objective function value is ∞.

12.4 Numerical results

Computational tests were performed on a PC with a 3GHz Pentium IV microprocessor and 1Gb of memory. We compare the original EM algorithm, as described in Sect. 12.2, with the herein proposed EM algorithm modified with the HJ *Local* procedure, described in Sect. 12.3. We use a collection of 18 test functions [2,3,5] (see Table 12.1).

Table 12.1. Test functions and the corresponding parameters used by EM

Test function	n	Box constraints	f_{global}	m	MaxIt	LSIt	δ
Shekel5	4	$[0,10]^4$	-10.153200	40	150	10	1.00E-03
Shekel7	4	$[0, 10]^4$	-10.402941	40	150	10	1.00E-03
Shekel10	4	$[0, 10]^4$	-10.536410	40	150	10	1.00E-03
Hartman3	3	$[0, 1]^3$	-3.862782	30	75	10	1.00E-03
Hartman6	6	$[0, 1]^6$	-3.322368	30	75	10	1.00E-03
Goldstein-Price	2	$[-2,2]^2$	3.000000	20	50	10	1.00E-03
Branin	2	$[-5,10]\times[0,15]$	0.397887	20	50	10	1.00E-03
Six-hump Camel	2	$[-3,3]\times[-2,2]$	-1.031628	20	50	10	1.00E-03
Shubert	2	$[-10, 10]^2$	-186.730909	20	50	10	1.00E-03
Griewank	2	$[-100, 100]^2$	0.000000	30	100	20	1.00E-03
Himmelblau	2	$[-6, 6]^2$	0.000000	10	50	5	1.00E-03
Sine envelope	2	$[-0.5, 0.5]^2$	0.000000	20	75	10	5.00E-04
Bohachevsky	2	$[-10, 10]^2$	0.000000	20	75	20	1.00E-03
Easom	2	$[-10, 10]^2$	-1.000000	20	50	10	1.00E-03
Hump	2	$[-5, 5]^2$	0.000000	20	50	10	1.00E-03
Spherical	2	$[-100, 100]^2$	0.000000	30	75	20	1.00E-03
Three-hump	2	$[-5, 5]^2$	0.000000	20	50	10	1.00E-03
Zakharov4	4	$[-5, 10]^4$	0.000000	30	75	20	1.00E-03

The first four columns of the table refer to the name of the function, the dimension of the problem, n, the default box constraints, and the known global optimum, f_{global}. The last four columns list the parameters used by EM for each function.

The results obtained by the original EM algorithm are shown in Table 12.2 and the results of the EM algorithm modified with the HJ *Local* procedure are presented in Table 12.3. We use average results for comparison, over 25 runs. Table 12.2 and Table 12.3 report average number of function evaluations, Ev_{avg}, the average best function values, f_{avg}, and the best function value, $f_{best} = \min (f^i_{best}, i = 1,\ldots,nruns)$, over $nruns$=25 runs. Values of the mean absolute error

$$\text{MAE} = \frac{|f_{avg} - f_{global}|}{n} \tag{12.6}$$

and the standard deviation

$$\text{SD} = \sqrt{\frac{\sum_{i=1}^{nruns} (f^i_{best} - f^i_{avg})^2}{nruns}} \tag{12.7}$$

are also listed for each problem.

Table 12.2. Results of original EM with *Local* procedure applied to the best point

Test function	Ev_{avg}	f_{avg}	f_{best}	MAE	SD
Shekel5*	1865	-10.152770	-10.153163	0.000108	0.000270
Shekel7	1480	-10.402471	-10.402884	0.000117	0.000486
Shekel10	1486	-10.535939	-10.536256	0.000118	0.000243
Hartman3	1260	-3.862487	-3.862722	0.000098	0.000111
Hartman6	1850	-3.322267	-3.322365	0.000017	0.000091
Goldstein Price	488	3.000141	3.000001	0.000071	0.000077
Branin	512	0.397906	0.397888	0.000009	0.000012
Six Hump Camel	291	-1.031599	-1.031628	0.000015	0.000026
Shubert	357	-186.723068	-186.730906	0.003920	0.005750
Griewank**	1582	0.000062	0.000008	0.000031	0.000049
Himmelblau	347	0.000047	0.000008	0.000023	0.000023
Sine envelope	518	0.000044	0.000002	0.000022	0.000027
Bohachevsky	793	0.000063	0.000001	0.000032	0.000041
Easom	489	-0.999966	-1.000000	0.000017	0.000029
Hump	287	0.000044	0.000001	0.000022	0.000027
Spherical	903	0.000033	0.000001	0.000017	0.000030
Three-hump	382	0.000041	0.000001	0.000021	0.000030
Zakharov4	1621	0.000068	0.000014	0.000017	0.000022

The termination criteria and the used parameters are the ones proposed in [2]. Thus, both algorithms stop when the number of iterations exceeds MaxIt unless the relative error in the best objective function value, with respect to f_{global}, is less than 0.01%. In the HJ algorithm, the factor used to reduce δ, whenever an unsuccessful iteration is found, is 0.1 and the minimum step length allowed was 1×10^{-8}. In the tables, * means that 4 (in Table 12.1) and 5 (in Table 12.2) runs of Shekel5 did not converge and ** means that 3 runs of Griewank did not converge.

The results obtained with the EM algorithm modified with the HJ *Local* procedure are better than the ones produced by the algorithm of Sect. 12.2, as far as the accuracy of the results is concerned. The proposed algorithm achieves in general the lowest numerical errors (MAE) and lowest standard deviations (SD) for 25 runs. However, Table 12.3 reveals in some cases larger number of function evaluations.

Table 12.3. Results of EM with HJ *Local* procedure applied to the best point

Test function	Ev_{avg}	f_{avg}	f_{best}	MAE	SD
Shekel5*	1880	-10.153169	-10.153186	0.000008	0.000023
Shekel7	1677	-10.402855	-10.402934	0.000022	0.000205
Shekel10	1679	-10.536320	-10.536405	0.000022	0.000177
Hartman3	1482	-3.862541	-3.862775	0.000080	0.000106
Hartman6	2378	-3.322308	-3.322361	0.000010	0.000075
Goldstein Price	405	3.000036	3.000001	0.000018	0.000054
Branin	372	0.397900	0.397888	0.000006	0.000012
Six Hump Camel	261	-1.031588	-1.031628	0.000020	0.000025
Shubert	641	-186.714293	-186.730827	0.008308	0.067471
Griewank**	2095	0.000018	0.000000	0.000009	0.000017
Himmelblau	294	0.000018	0.000000	0.000009	0.000021
Sine envelope	423	0.000025	0.000000	0.000012	0.000030
Bohachevsky	1892	0.000018	0.000000	0.000009	0.000011
Easom	476	-0.999965	-1.000000	0.000018	0.000033
Hump	315	0.000023	0.000000	0.000012	0.000032
Spherical	514	0.000031	0.000000	0.000015	0.000032
Three-hump	337	0.000027	0.000001	0.000013	0.000021
Zakharov4	1430	0.000013	0.000001	0.000003	0.000027

12.5 Conclusions

We have studied the Electromagnetism-like algorithm and implemented a new *Local* procedure based on a pattern search method. This new algorithm was applied to different test problems from the literature and com-

pared with the original EM. The preliminary results seem promising. Future developments will focus on extending the numerical experiments to other test functions with larger dimensions and many local minimizers in the feasible region. The purpose here is to analyze the pattern search ability to drive the best point towards the global minimizer instead of to a non-global one.

References

1. Birbil SI (2002) Stochastic global optimization techniques. PhD Thesis, North Carolina State University
2. Birbil SI, Fang S (2003) An electromagnetism-like mechanism for global optimization. J. of Global Optimization 25(3):263-282
3. Dixon LCW, Szegö GP (1978) The global optimization problem: an introduction. In: Dixon LCW, Szegö GP (eds) Towards Global Optimisation 2. North-Holland, Amsterdam, pp 1-15
4. Hooke R, Jeeves TA (1961) Direct search solution of numerical and statistical problems. J. Assoc. Comput. Mach. 8:212-229
5. Huyer W, Neumaier A (1999) Global optimization by multilevel coordinate search. J. Global Optimization 14:331-355
6. Lewis RM, Torczon V (1999) Pattern search algorithms for bound constrained minimization. SIAM J. Optim. 9(4):1082-1099