

## A Bacterial Chemotaxis Multiobjective Optimization Algorithm

Maria Alejandra Guzmán<sup>1</sup>, Alberto Delgado<sup>2</sup>, Jonas de Carvalho<sup>3</sup>

<sup>1</sup>University of São Paulo- National University of Colombia, carrera 30 # 45-03, Bogotá, Colombia, maguzmanp@unal.edu.co; <sup>2</sup>National University of Colombia, carrera 30 # 45-03, Bogotá, Colombia, adelgado@ieee.org; <sup>3</sup>University of São Paulo, Av. Trabalhador São Carlense, 400, São Carlos, Brazil, prjonas@sc.usp.br

### 1. Abstract

This paper presents a multiobjective optimization algorithm based on bacterial chemotaxis BCMOA (Bacterial Chemotaxis Multiobjective Optimization Algorithm), it incorporates chemotactical responses of individuals as a viable bioinspired optimization technique. As other multiobjective optimization algorithms, the BCMOA works with populations, in this case bacterial colonies, which are located into the search space. The locations represent the value of the objective functions for possible solutions to the optimization problem, thus the well known non-dominated sorting concept is incorporated in order to classify the location of each bacterium. The bacterial chemotactic behavior is exploited by using the ability to sense the environmental conditions and make temporal-space comparisons to the actual and previous locations, always with the goal of finding nutrient sources (optimal locations) and escape from noxious environments (dominated locations). The self organized collective behavior showed for bacterial colonies through communications and signaling between bacteria are also considered. The bacterium motile behavior is emulated by application of simple responses to the signal sense in a way that each bacterium develops one of two possible chemotactical actions, swimming or tumbling. The proposed algorithm is validated using the standard test problems SCH, FON, POL, KUR, ZDT1 to ZDT4, and different performance measures were also implemented to compare the performance of BCMOA with the popular algorithm NSGA-II.

**2. Keywords:** multiobjective optimization, bacterial chemotaxis, bioinspired methods.

### 3. Introduction

Designing an engineering product is not an easy task. In a globalised market the demands and needs of customers are becoming more sophisticated and extremely changing. The speed of change along with the strong competition requires innovative products of high quality at low prices. Consequently, the development of a product is a process of increasing complexity that demands permanent improvements that lead to the efficient use of resources. The importance of optimizing the design of a product justifies the permanent search for more effective and robust optimization techniques.

The emulation of nature has inspired scientists in various fields through the history of mankind. Recently, due to advances in computing and the rise of ideas extracted from behaviors of living beings and natural processes, the techniques inspired in nature have gained interest motivated mainly by two causes [1]:

- Traditional methods have proven to be incapable of adequately handling complex problems, characterized by the manipulation of a large number of variables and the lack of complete mathematical models.
- For most complex engineering problems there is a similar version in nature.

In the year 2005, Amos, Hodgson and Gibbons published an article [2] in which they exposed the potential of exploring bacterial chemotaxis as a distributed optimization process. The paper highlighted the fact that existing algorithms so far do not consider the interaction and communication between bacteria as happens in a natural colony. Knowing that it is precisely this collective behavior that enables them to develop biologically advantageous patterns, the authors suggested the possible application of this interaction between individuals in the development of distributed optimization methods. In this research an application of chemotaxis and pattern formation in bacterial colonies is developed to solve multiobjective optimization problems.

### 4. Bacterial Chemotaxis

In a general sense, chemotaxis is the chemically directed movement, developed by some living beings. This movement and the chemical substances involved in it are used by some unicellular organisms, insects and even by the human being for different purposes, for example, to search for nutrients, to avoid predators, to generate communication between individuals, in the formation of colonies or groups, for sexual attraction or for territorial demarcation [3]. Beyond this broad definition, in the scientific literature the word chemotaxis is almost always used for cell movement in response to gradients of chemical concentrations present in the environment.

A bacterium is a prokaryotic unicellular organism, it does not have organelles linked to the cell membrane nor a real nucleus and its DNA is only a circular molecule. Its structure is basically conformed by a central body of microscopic size that can take many

different forms [4] and whose size can vary from  $0.01 \mu\text{m}^3$  as in the case of *Pelagibacter ubique* [5] to a volume  $10^{10}$  times bigger as in the case of *Epulopiscium fishelsoni* [6]. The cell wall is a rigid structure formed with molecules of polysaccharides limited by aminoacids that cover the cell membrane. Many bacteria are endowed with a series of rotating flagella in its cell surface that act as propellants to allow them to move at a speed of up to  $30 \mu\text{m}$  per second [7]. In addition to the appropriate structure to move in an autonomous way, bacteria have potent receivers located at the cell surface called chemoreceptors, capable of detecting temporal-space changes of chemical concentrations in the environment that surrounds them. In this way, when an external perturbation is detected, bacteria change their movement from a random walk to a biased random walk.

After being neglected for years, the research on bacterial chemotaxis began in the 60s starting with the work of Adler [8]. Among all bacteria *E. Coli* and *Salmonella typhimurium* have been the most studied and a large number of references to them can be found in the scientific literature [9-11]. Other types of bacteria and cells have also been the subject of study, for example, the *Pseudomonas putida* in [12], the *Thiovulum majus* in [13], the *Myxococcus xanthus* in [14], the *Deleya marina* in [15] and the cell of the human blood, *neutrophil* [16].

Chemotaxis is part of the overall survival strategy which was abbreviated by Young [17]:

“Bacteria want what all other organisms want: to grow, they need to eat; to reproduce, they need to divide; if things are good where they are, they want to stay; if things are better somewhere else, they want to move; if threatened, they need to escape; and if the world around them changes, they must change.”

Using their receptors a bacterium detects chemicals in its environment and makes a temporal-space comparison of the gradients found. In other words a bacterium uses its memory [18] and depending on the external conditions makes a decision and rotates the flagella doing a tumble or swimming.

#### **Swimming:**

Bacterial flagella have a helical shape, they can rotate at high speeds (270 or 600 rps) [19], they can stop momentarily and change the direction of rotation in a controlled manner [20]. As a result when they all rotate in the same direction, act as propellants moving the bacterium gently forward in an almost rectilinear movement.

#### **Tumbling:**

When bacterial flagella rotate in the opposite direction of swimming, they destabilize due to its helical shape propelling the bacterium in different directions at the same time, so the bacterial body tumbles randomly.

Using different combinations of these two movements, varying the length and duration, bacterium moves in its environment.

### **4.1. Bacterial chemotaxis as optimization process**

As can be deduced intuitively bacterial chemotaxis is an optimized strategy of movement so its application as an optimization strategy is not an entirely novel idea. The first one to raise this biological process as an optimization algorithm was Bremermann [21].

Considering a nonhomogeneous concentration of an attractor the algorithm proposed by Bremermann consists of the following steps:

- 1) The bacterium tumbles and dashes off in a straight line which is a random direction.
- 2) If the direction is one of increasing concentration, the bacterium continues until the concentration starts to decline. After sensing the decline, the bacterium tumbles again and as a result dashes off in a new random direction.
- 3) If the new direction is one of declining concentration of the attractant, then the swimming stops, the bacterium tumbles for a while and dashes off in a new random direction.
- 4) It is assumed that successive random directions are independent and uniformly distributed.

The results presented by Bremermann were the basis for two further developments, in [22] a chemotaxis optimization algorithm was used to maximize a profit function for fed-batch bioreactors. In [23] a new algorithm was developed giving importance to the concentration so bacterium does not stay swimming in the same direction when it finds positive gradients. The algorithm was applied to the solution of inverse airfoil design.

Other works have been developed by adapting the process of bacterial chemotaxis to the solution of optimization problems. In [24] it was developed an optimization algorithm in which besides considering the locomotion of bacteria, mechanisms as reproduction, dispersion and death acquire great importance. In this work, the communication between bacteria influence the chemotaxis process, getting closer to the concept that chemotaxis is a phenomenon of a bacterial colony rather than an individual behavior. In [25] two new methods for distributed nongradient optimization were presented and tested. For *E. Coli* the proposed algorithm is based on bacterial chemotaxis and for the *M. Xanthus* the basis of the optimization algorithm is the social foraging behavior. In [26] inspired by the work of Passino, it was developed an algorithm that simulates bacterial chemotaxis applying cellular automata and considering diffusion processes of chemical signals.

## 5. Multiobjective Optimization (MOO)

Most real world optimization problems require to make decisions involving more than one goal, when these goals are the minimization or maximization of functions we are referring to multi-objective optimization. A Multiobjective Optimization Problem (MOOP) is defined as the problem of finding a vector of decision variables that satisfies some restrictions and optimize a vector function whose elements represent the values of the functions. A MOOP may be formulated as follows [27]:

$$\begin{aligned} \text{Maximize / minimize:} \quad & f_m(x) & m = 1, 2, \dots, M \\ \text{Subject to:} \quad & g_j(x) \geq 0 & j = 1, 2, \dots, J \\ & h_k(x) = 0 & k = 1, 2, \dots, K \\ & x_i^L \leq x_i \leq x_i^U & i = 1, 2, \dots, n \end{aligned}$$

Where  $x$  is the vector of decision variables  $x = (x_1, x_2, \dots, x_n)^T$  and  $f_m(x)$  are the  $m$  objective functions. The values  $x_i^L$  and  $x_i^U$  represent respectively the minimum and maximum acceptable values for the variable  $x_i$ . These values define the boundary of the search space. The  $J$  inequalities  $g_j$  and the  $K$  equalities  $h_k$  are known as constrain functions.

In multiobjective optimization exist a set of solutions which after evaluation of non dominance conditions result to be non dominated solutions, that is why the notion of optimality refers to a set of optimal solutions instead of a single optimal. This set of optimal solutions is known as Optimal Pareto Front. In most cases it is not easy to find the analytical expressions of the line or curve that contains the optimal solutions which typically are calculated the Pareto points and the objective functions values in them. Due to these conditions, there are two goals that any Multiobjective Optimization Algorithm (MOOA) seeks to achieve [28]:

- Guide the search towards the global Pareto Optimal region
- Maintain population diversity in the Pareto Optimal Front.

Since the 50s in the area of operational research a variety of methods known as classical, has been developed for the solution of MOOP. Those methods are based on formal logic or mathematical programming. Some of the most representative classical methods are linear programming [29], the weighted sum method and the Goal Programming method [30].

Computation inspired by nature, that aims to use ideas extracted from nature to develop computational tools and implement techniques [31], has not been absent from the discussion in the area of MOO. Among bioinspired optimization techniques, the most famous is Genetic Algorithms (AG's). The pioneering work in the practical application of the fundamentals of AG's to multiobjective optimization is the VEGA (Vector Evaluated Genetic Algorithm) [32]. At present the most popular Genetic Algorithm for solving MOO is the NSGA-II (Elitist Non-dominated Sorting Genetic Algorithm) [33] which was used in this work as a reference to compare the performance of the proposed chemotaxis based algorithm. Another bioinspired approach is the so called PSO (Particle Swarm Optimization) which was recently implemented in the solution of MOOP with algorithms such as MOPSO (Multiobjective Particle Swarm Optimization) [34], NSPSO (Non-dominated Sorting Particle Swarm Optimizer) [35] and TV-MOPSO (Time Variant Multi-Objective Particle Swarm Optimization) [36]. Although bacterial chemotaxis has been already implemented as an optimization process, it has not been applied in the area of multiobjective optimization.

In multiobjective optimization, an important argument in favour of classical methods over bioinspired is the existence of theorems that prove convergence towards an optimal solution. However there are some fundamental advantages of algorithms inspired by nature (often based on populations) over classic algorithms:

- The capability to work in a distributed way in order to find several optimal solutions in a single run, classical methods can only find a solution at a time that may not be an optimal solution so it is necessary to run the classical algorithm many times to find a different optimal solution every run.
- Bioinspired methods are less sensitive to the shape of the Pareto Optimal Front than classical methods, which present difficulties to solve some specific types of multiobjective optimization problems, for example, when the Optimal Pareto Front is nonconvex or disconnected [37].
- Most of the classical methods work with several objectives by fusing them into a single objective, as a result the optimal solutions found for this problem have a strong dependence of the parameter selection made by the user [38].

In the following the BCMOA algorithm is presented, this algorithm uses fast nondominated sorting procedure, communication and information exchange between the colony members and a simple chemotactic strategy to change the bacterial positions in order to explore the search space to find several solutions on the Optimal Pareto Front.

## 6. Bacterial Chemotaxis Multiobjective Optimization Algorithm (BCMOA)

*Initializing:*

In BCMOA the bacteria locations represent the value of the decision variables, so the search space is limited for the variables range. The objective functions values represent the amount of nutrients present in the environment. All bacteria from a colony with  $S$  members are located initially at random positions and then the conditions of the environment in those positions are sensed by every

bacterium.

*Objective evaluation and non dominance classification:*

Applying a fast nondominated sorting procedure [33] the current locations are classified in Optimal Pareto Fronts. Those bacteria whose location represent a nondominated solution, are classified within the first set of Optimal Pareto Front (OPF1) and are called “strong bacteria” because their environment is rich in nutrients, so they have a sufficient amount of nutrients to eat. The remaining bacteria are classified into different OPF sets according to their nondominance condition and are called “weak bacteria” seeking to represent the bacteria that do not have enough food.

*Chemotactical strategy for strong bacteria:*

Making use of their temporal-space memory, the strong bacteria compare the nondominance classification of their current location (OPF) with the previous and as a result of these comparisons each strong bacterium react with any of the two possible movements: if both the previous and the current location are rich in nutrients (OPF1), the bacterium takes a very small step in a random direction (tumble) without straying far from the current source of nutrients. On the other hand if only the current location is rich in nutrients, the bacterium takes a bigger tumble.

*Communication and chemotactical strategy for weak bacteria:*

The strong bacteria send a signal to the weak bacteria indicating that in their position there is a source of nutrients. The weak bacteria take advantage of this exchange of information as follows: each weak bacteria knows that if it goes in the direction of a strong bacteria location will find a source of nutrients, so each weak bacteria select randomly a strong bacteria, goes to its rich location and keeping the same direction take a swimming step besides the rich location.

Fig. 1 illustrates schematically the chemotactical behavior of the colony for a multiobjective optimization problem with two objective functions. The black curve represents the Optimal Pareto Front, black spots represent strong bacteria and white spots represent weak bacteria, both of them before the chemotactic response. Gray spots represent all bacteria after one step. The figure also represents three different steps in the chemotactic strategy: long tumble, short tumble and swim.

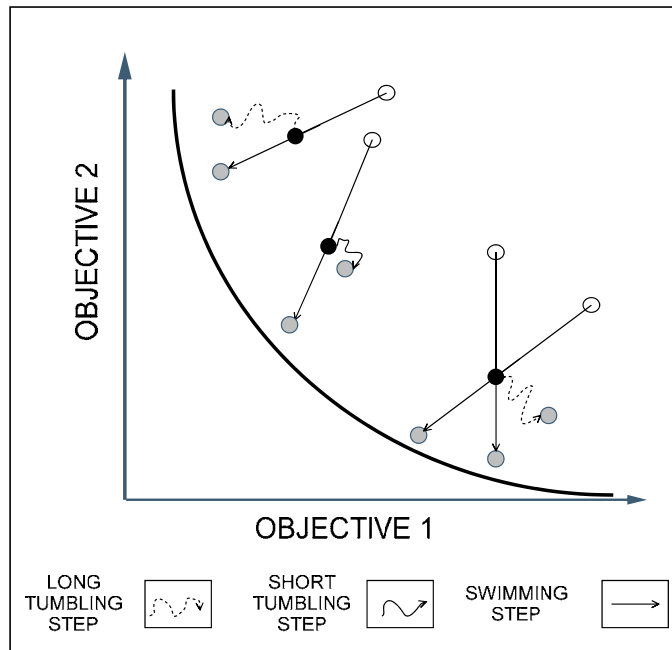


Fig. 1. Chemotactic behavior in BCMOA

BCMOA can be summarized in the following steps:

1. Define the parameters  $S$ ,  $CHS_{max}$ ,  $Sup$ ,  $Frac_1$ ,  $Frac_2$ , and  $Frac_3$ . Initialize the population.  $t = 0$ .
  - a) The current location of the  $i$ -th bacterium  $L_i$  is initialized with random real numbers within the specified decision variable range.
  - b) Store all bacteria and their current location in a list Bac.
  - c) Actualize the parameters FACTOR, STS, LTS and SS, according to Eq.(1), Eq.(2), Eq.(3) and Eq.(4).  $t = t + 1$ .

$$FACTOR = (CHS_{max} - t) / CHS_{max} \tag{1}$$

$$STS = Frac_1 * FACTOR \tag{2}$$

$$LTS = Frac_2 * FACTOR \quad (3)$$

$$SS = Frac_3 * FACTOR \quad (4)$$

2. Evaluate each bacterium in the list Bac.
3. Classify the location of all bacteria using non-dominance criterium and store each bacterium, its location and its Optimal Pareto Front classification in a list OPFclas. For  $t = 1$   $OPFclasPrev = OPFclas$ .
4. Identify the bacteria whose current locations were classified in the OPF1 (strong bacteria). Store the strong bacteria and its location in a list Bacnondom. Store the other bacteria of the colony and their locations in a list Bacdom.
5. For each bacterium in Bacnondom check the previous OPF classification wich is stored in OPFclasPrev.
6. Identify each bacterium  $i$  in Bacnondom whose previous OPF classification was 1; create a unitary vector in a random direction  $UV_{rand}$  and actualize the locations of these bacteria according to Eq. (5):

$$L_i(t+1) = L_i(t) + UV_{rand_i} * STS \quad (5)$$

7. For each bacteria  $i$  in Bacnondom whose previous OPF classification was not 1, create a unitary vector in a random direction  $UV_{rand}$ . Actualize the location of these bacteria according to Eq (6):

$$L_i(t+1) = L_i(t) + UV_{rand_i} * LTS \quad (6)$$

8. For each bacteria  $i$  in Bacdom, select randomly a bacterium  $j$  from Bacnondom, create a unitary vector  $UV_{dir}$  in the direction from the location of the weak bacterium  $i$  to the location of the strong selected bacteria  $j$ . Actualize the location of each weak bacteria according to Eq (7):

$$L_i(t+1) = L_j(t) + UV_{dir_i} * SS \quad (7)$$

9. Apply an strategy of Absorbing Walls [39]: When a bacterium hits the boundary of the solution space in one of the dimensions, its location is pulled back to the allowed solution space in that dimension.
10.  $OPFclasPrev = OPFclas$ .
11. If  $t < CHS_{max}$  go to 1.b.
12. Obtain the Optimal Pareto Front from locations of last population in Bac.

## 7. Experiments

The effectiveness of BCMOA was validated using eighth standard test problems and two performance measures, all of them selected according to DEB [33]. The performance of BCMOA has been compared with the NSGA-II, both algorithms were implemented on Matlab 7.

### 7.1 Test problems

The parameters used for each test problem are given in Table 1. Table 2 describes the test problems used to evaluate the performance of the BCMOA, in all problems the objective function must be minimized. For all problems in both algorithms the inicial population was 100 individuals. For SCH, FON, POL and KUR problems 100 iterations were made and for the problems of ZDT series, 250 iterations were made.

Table 1. Parameters of test problems for BCMOA

	S	$CHS_{max}$	$Sup$	$Frac_1$	$Frac_2$	$Frac_3$
SCH	100	100	$10^3$	$1 \times 10^{-6}$	$1 \times 10^{-3}$	$1 \times 10^{-2}$
FON	100	100	4	$1 \times 10^{-6}$	$1 \times 10^{-2}$	$1 \times 10^{-1}$
POL	100	100	$\pi$	$1 \times 10^{-6}$	$1 \times 10^{-2}$	$1 \times 10^{-1}$
KUR	100	100	5	$1 \times 10^{-6}$	$1 \times 10^{-2}$	$1 \times 10^{-1}$
ZDT1	100	250	1	$1 \times 10^{-6}$	$1 \times 10^{-1}$	1
ZDT2	100	250	1	$1 \times 10^{-6}$	$1 \times 10^{-1}$	1
ZDT3	100	250	1	$1 \times 10^{-6}$	$1 \times 10^{-1}$	1
ZDT4	100	250	5	$1 \times 10^{-6}$	0.01 for $i = 1$ , 0.1 for $i = 2 : 10$	0.1 for $i = 1$ , 1 for $i = 2 : 10$

For NSGA-II implementation as suggested in [33] were used SBX (simulated binary crossover) operator, polynomial mutation, a crossover probability of  $p_c = 0.9$  and a mutation probability of  $p_m = 1/n$  where  $n$  is the number of decision variables.

### 7.2 Performance measures

We implemented the diversity metric  $\Delta$  which measures the extent of spread achieved among the obtained solutions and the Convergence metric  $\Upsilon$  wich measures the extent of convergence to a known set of Optimal Pareto solutions. A detailed explanation of these metrics can be found at [33].

Table 2. Test problems for the BCMOA

Problem	$n$	Variable bounds	Objective Functions	Optimal Solutions
SCH	1	$[-10^3, 10^3]$	$f_1(x) = x^2$ $f_2(x) = (x-2)^2$	$x \in [0, 2]$
FON	3	$[-4, 4]$	$f_1(x) = 1 - \exp\left(-\sum_{i=1}^3 \left(x_i - \frac{1}{\sqrt{3}}\right)^2\right)$ $f_2(x) = 1 - \exp\left(-\sum_{i=1}^3 \left(x_i + \frac{1}{\sqrt{3}}\right)^2\right)$	$x_1 = x_2 = x_3 \in \left[-\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}\right]$
POL	2	$[-\pi, \pi]$	$f_1(x) = \left[1 + (A_1 - B_1)^2 + (A_2 - B_2)^2\right]$ $f_2(x) = \left[(x_1 + 3)^2 + (x_2 + 1)^2\right]$ $A_1 = 0.5 \sin(1) - 2 \cos(1) + \sin(2) - 1.5 \cos(2)$ $A_2 = 1.5 \sin(1) - \cos(1) + 2 \sin(2) - 0.5 \cos(2)$ $B_1 = 0.5 \sin(x_1) - 2 \cos(x_1) + \sin(x_2) - 1.5 \cos(x_2)$ $B_2 = 1.5 \sin(x_1) - \cos(x_1) + 2 \sin(x_2) - 0.5 \cos(x_2)$	
KUR	3	$[-5, 5]$	$f_1(x) = \sum_{i=1}^{n-1} \left(-10 \exp\left(-0.2 \sqrt{x_i^2 + x_{i+1}^2}\right)\right)$ $f_2(x) = \sum_{i=1}^n \left( x_i ^{0.8} + 5 \sin(x_i^3)\right)$	
ZDT1	30	$[0, 1]$	$f_1(x) = x_1$ $f_2(x) = g(x) * \left[1 - \sqrt{\frac{x_1}{g(x)}}\right]$ $g(x) = 1 + 9 * \frac{\left(\sum_{i=2}^n x_i\right)}{(n-1)}$	$x_1 \in [0, 1]$ $x_i = 0$ $i = 2, \dots, 30$
ZDT2	30	$[0, 1]$	$f_1(x) = x_1$ $f_2(x) = g(x) * \left[1 - \left(\frac{x_1}{g(x)}\right)^2\right]$ $g(x) = 1 + 9 * \frac{\left(\sum_{i=2}^n x_i\right)}{(n-1)}$	$x_1 \in [0, 1]$ $x_i = 0$ $i = 2, \dots, 30$
ZDT3	30	$[0, 1]$	$f_1(x) = x_1$ $f_2(x) = g(x) * \left[1 - \sqrt{\frac{x_1}{g(x)}} - \frac{x_1}{g(x)} \sin(10\pi x_1)\right]$ $g(x) = 1 + 9 * \frac{\left(\sum_{i=2}^n x_i\right)}{(n-1)}$	$x_1 \in [0, 1]$ $x_i = 0$ $i = 2, \dots, 30$
ZDT4	10	$x_1 \in [0, 1]$ $x_i \in [-5, 5]$	$f_1(x) = x_1$ $f_2(x) = g(x) * \left[1 - \sqrt{\frac{x_1}{g(x)}}\right]$ $g(x) = 1 + 10(n-1) + \sum_{i=2}^n \left[x_i^2 - 10 \cos(4\pi x_i)\right]$	$x_1 \in [0, 1]$ $x_i = 0$ $i = 2, \dots, 10$

## 8. Results and Discussion.

Results reported in Table 3 and Table 4 are the mean (first row) and the variance (second row) values of the diversity metric and convergence metric respectively over 10 simulations of the BCMOA and the NSGA-II algorithms for SCH, FON, ZDT1, ZDT2, ZDT3 and ZDT4. Table 5 presents the mean of the execution time in seconds over 10 simulations of both algorithms for FON, KUR and ZDT3.

Table 3. The Diversity metric  $\Delta$

		SCH	FON	ZDT1	ZDT2	ZDT3	ZDT4
BCMOA	Mean	0.7651	0.5343	0.5182	0.5266	0.8455	0.9541
	Variance	0.002576	0.000809	0.001468	0.0010	0.0008	0.0003
NSGA II	Mean	0.3825	0.4141	0.4064	0.4395	0.7241	1.0140
	Variance	0.001073	0.000989	0.001269	0.0011	0.0005	0.0109

Table 4. The Convergence metric  $\gamma$

		SCH	FON	ZDT1	ZDT2	ZDT3	ZDT4
BCMOA	Mean	0.0033	0.0032	0.0143	0.0133	0.0081	20.3709
	Variance	4.36E-08	1.89E-07	2.13E-05	6.74E-05	1.80E-06	44.6
NSGA II	Mean	0.0031	0.0023	0.0040	0.0027	0.0054	10.2906
	Variance	4.64E-08	1.24E-08	3.14E-07	7.69E-08	1.00E-07	47.6

Table 5. Execution time

		FON	KUR	ZDT3
BCMOA	Mean	18.5250	19.8328	65.2218
NSGA II	Mean	276.0653	251.0156	634.8984

Figures 2 to 7 show the resultant non-dominated fronts corresponding to the test problems with BCMOA.

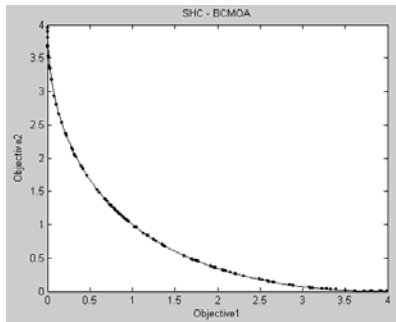


Fig. 2. OPF for SCH with BCMOA.

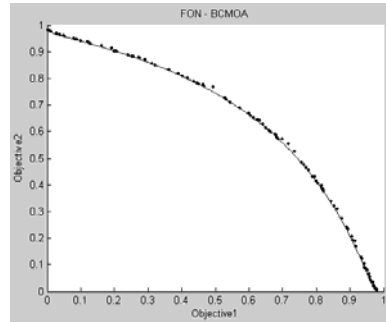


Fig. 3. OPF for FON with BCMOA.

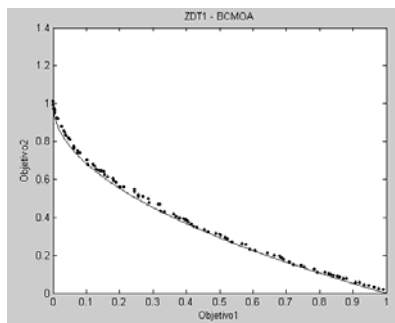


Fig. 4. OPF for ZDT1 with BCMOA.

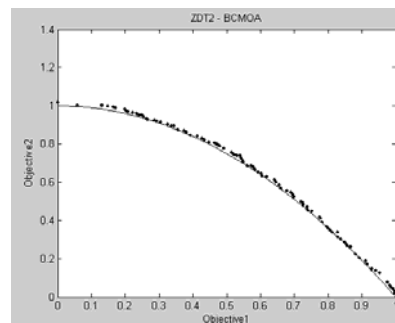


Fig. 5. OPF for ZDT2 with BCMOA.

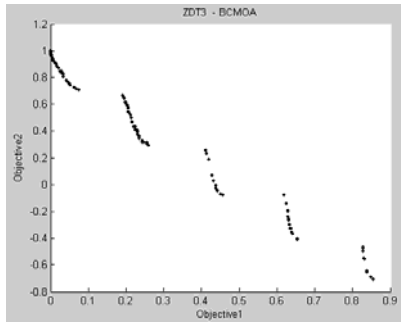


Fig. 6. OPF for ZDT3 with BCMOA.

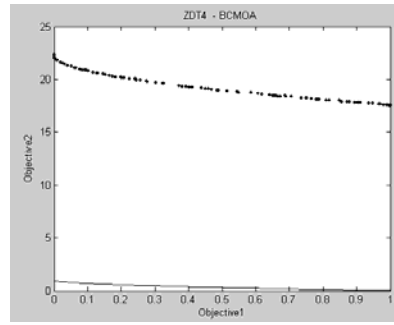


Fig. 7. OPF for ZDT4 with BCMOA.

Figures 8 to 11 show the resultant non-dominated fronts corresponding to POL and KUR test problems with BCMOA and NSGA-II.

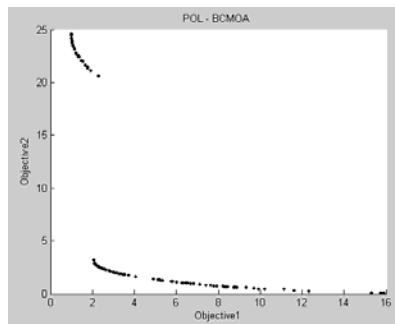


Fig. 8. OPF for POL with BCMOA.

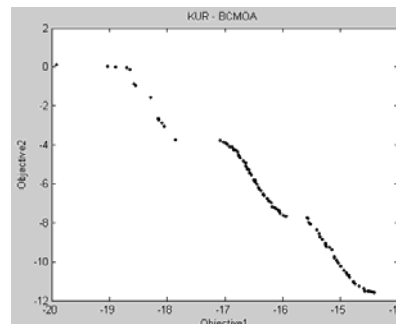


Fig. 9. OPF for KUR with BCMOA.

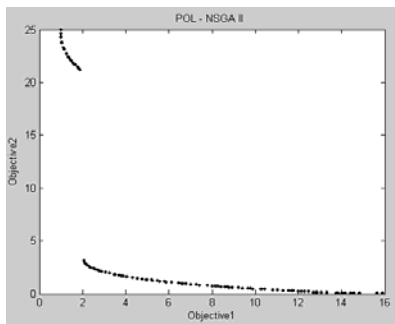


Fig. 10. OPF for POL with NSGA-II.

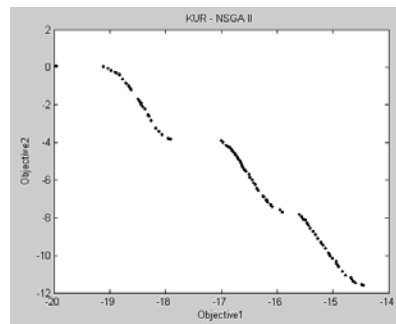


Fig. 11. OPF for KUR with NSGA-II.

For all test problems, except for ZDT4, the BCMOA found solutions closer to the OPF with an acceptable diversity. For all problems the last population belongs entirely to the same OPF. The results of the performance measures show that BCMOA has worse convergence and diversity compared to the NSGA-II. The execution times for the BCMOA compared with the NSGA-II are 14.9 times lower for FON problem, 12.65 times lower for KUR problem and 9.73 times lower for ZDT3 problem.

## 9. Conclusion and Future Work

We have presented a new multiobjective optimization algorithm introducing a novel approach based on bacterial chemotaxis. For seven different difficult test problems the BCMOA satisfactorily meets the two goals of MOO because not only was effective in finding solutions towards the OPF but also the diversity of the solutions on the front found was good.

Although the performance measures show that BCMOA has not given the best results compared to the NSGA-II, the BCMOA shows its high potential when the simplicity of the algorithm is considered. The BCMOA does not apply process of diversity preservation, density estimation or crowded-comparison as the NSGA-II, making its implementation extremely simple. Execution times also reflect the low computational complexity of the algorithm, observing the big differences between the two algorithms BCMOA maintains its competitiveness.

Future research could develop strategies to improve the diversity of solutions. Also a constraint handling approach could be formulated and tested in the solution of standard constrained multiobjective optimization problems.

## 10. Acknowledgements

The first author acknowledges financial support from the National University of Colombia (Universidad Nacional de Colombia) and support from the National Counsel of Technological and Scientific Development (Conselho Nacional de Desenvolvimento Científico e Tecnológico – CNPq) through a scholarship to pursue doctoral studies at University of São Paulo.

## 11. References

1. De Castro, L.; Von Zuben, F. Recent developments in biologically inspired computing. Hershey, PA, USA: Idea Group Publishing, 2004.
2. Amos, M., Hodgson, D, Gibbons, A. Bacterial self-organisation and computation. Submitted to International Journal of Unconventional Computing, 2005, <http://arxiv.org/abs/q-bio/0512017>.
3. Murray, J. *Mathematical Biology: An Introduction*. Secaucus, NJ, USA:Springer-Verlag New York, 2002.
4. Young, K. The selective value of bacterial shape. *Microbiology and Molecular Biology Reviews*, 2006, 70(3), 660-703.
5. Rappe, M.S., et al. Cultivation of the ubiquitous SAR11 marine bacterioplankton clade. *Nature*, 2002, 418(6898), 630-633.
6. Angert, E.R., Clements, K., Pace, N. The largest bacterium. *Nature*, 1993, 362(6417), 239-241.
7. Eisenbach, M, et al. *Chemotaxis*. Singapore: Imperial College Press, 2004.
8. Adler, J. Chemotaxis in bacteria. *Science*, 1966, 153, 708-716.
9. Berg, H.C., Brown, D. Chemotaxis in *Escherichia Coli* analyzed by three-dimensional tracking. *Nature*, 1972, 239(5374), 500-504.
10. Sourjik, V. Receptor clustering and signal processing in *E. Coli* chemotaxis. *Trends in Microbiology*, 2004, 12(12), 569-576.
11. Polezhaev, A, et al. Spatial patterns formed by chemotactic bacteria *Escherichia Coli*. *International Journal of Developmental Biology*, 2006, 5(2-3), 309-314.
12. Hilpert, M. Lattice-Boltzmann model for bacterial chemotaxis. *Mathematical Biology*, 2005, 51(3), 302-332.
13. Thar, R., Kuhl, M. Complex pattern formation of marine gradient bacteria explained by a simple computer model. *FEMS Microbiology Letters*, 2005, 246(1), 75-79.
14. Mauriello, E., Zusman, D. Polarity of motility systems in *Myxococcus Xanthus*. *Current Opinion in Microbiology*, 2007, 10(6), 624-629.
15. Yagi, H., Maruyama, A. Novel diglycosyldiacylglycerol from the gram-negative bacterium *Deleya marina*. *Biochimica et Biophysica Acta (BBA) – Lipids and Lipid Metabolism*, 1998, 1393(1), 161-165.
16. Onsum, M., Arkin, A. Autonomous mobile robot control based on white blood cell chemotaxis. *Computational Methods in Systems Biology*, 2005, 3082, 9-19.
17. Young, K. The selective value of bacterial shape. *Microbiology and Molecular Biology Reviews*, 2006, 70(3), 660-703.
18. Segall, J., Block, S., Berg, H. Temporal comparisons in bacterial chemotaxis. *Proceedings of the National Academy of Sciences of the United States of America*, 1986, 83(23), 8987-8991.
19. Kudo, S., Magariyama, Y., Aizawa, S.I. Abrupt changes in flagellar rotation observed by laser dark-field microscopy. *Nature*, 1990, 346(6285), 677-680.
20. Eisenbach, M. Functions of the flagellar modes of rotation in bacterial motility and chemotaxis. *Molecular Microbiology*, 1990, 4(2), 161-167.
21. Bremermann, H. Chemotaxis and optimization. *J Franklin Institute*, 1974, 297, 397-404.
22. Montague, G.A., Wardb, A.C. A Sub-optimal solution to the optimisation of bioreactors using the chemotaxis algorithm. *Process Biochemistry*, 1994, 29(6), 489-496.
23. Müller, S.D., et al. Optimization based on bacterial chemotaxis. *IEEE Transactions on Evolutionary Computation*, 2002, 6(1), 16-29.
24. Passino, K. Biomimicry of bacterial foraging for distributed optimization and control. *IEEE Control Systems Magazine*, 2002, 22(3), 52-67.
25. Liu, Y., Passino, K. Biomimicry of social foraging bacteria for distributed optimization: models, principles, and emergent behaviors. *Journal of Optimization Theory and Applications*, 2002, 115(3), 603-628.
26. Kim, T.H., Jung, S.H., Cho K.H. Investigations into the design principles in the chemotactic behavior of *Escherichia Coli*. *BioSystems*, 2007, 91(1), 171-182.
27. Deb, K. *Multi-objective optimization using evolutionary algorithms*. New York: John Wiley & Sons, 2001.
28. Deb, K. Evolutionary algorithms for multi-criterion optimization in engineering design. *Proceedings of Evolutionary Algorithms in Engineering and Computer Science (EUROGEN-99)*, 1999, 29 may- 03 June, Jyväskylä, Finlandia, 135-161.
29. Dantzig, G., Thapa, M. *Linear Programming: Introduction*. New York: Springer-Verlag New York Incorporated, 1997.
30. Ehrgott, M. (Ed.). *Multiple criteria optimization: state of the art. Annotated bibliographic surveys*. Secaucus, NJ, USA: Kluwer Academic Publishers, 2002.
31. De Castro, L. Fundamentals of natural computing: an overview. *Physics of Life Reviews*, 2007, 4(1), 1-36.
32. Schaffer, J.D. Some experiments in machine learning using vector evaluated genetic algorithm (artificial intelligence, optimization, adaptation, pattern recognition). 166, PhD. These, Vanderbilt University, Nashville, USA, 1984.
33. Deb, K., et al. A fast and elitist multiobjective genetic algorithm: NSGA II. *IEEE Transactions on Evolutionary Computation*, 2002, 6(2), 182-197.
34. Coello, C.A.C., Toscano, G., Salazar, M. Handling multiple objectives with Particle Swarm Optimization. *IEEE Transactions on Evolutionary Computation*, 2004, 8(3), 256-279.
35. Li, X. A non-dominated sorting particle swarm optimizer. *Proceeding of Genetic and Evolutionary Computation Conference 2003 (GECCO'03)*, Lecture Notes in Computer Science, 2003, Chicago, USA, 37-48.
36. Kumar, P., Bandyopadhyay, S., Kumar, S. Multi-objective particle swarm optimization with time variant inertia and

- acceleration coefficients. *Information Sciences*, 2007, 177(22), 5033-5049.
37. Coello, C.A.C. Evolutionary multiobjective optimization: A historical view of the field. *IEEE Computational Intelligence Magazine*, 2006, 1(1), 28-36.
  38. Sarker, R. (Ed.). *Evolutionary Optimization*. Secaucus, NJ, USA: Kluwer Academic Publishers, 2002
  39. Robinson, J., Rahmat-Samii, Y. Particle swarm optimization in electromagnetics. *IEEE Transactions on Antennas and Propagation*, 2004, 52(2), 397-407.