BACTERIAL FORAGING OPTIMIZATION ALGORITHM APPLIED TO ENGINEERING SYSTEM DESIGN

Fran Sérgio Lobato, fslobato@mecanica.ufu.br
Valder Steffen Jr, vsteffen@mecanica.ufu.br
School of Mechanical Engineering, FEMEC,
Universidade Federal de Uberlândia, UFU, P.O. Box 593, 38400-902, Uberlândia-MG, Brazil.

Abstract. *Nowadays, optimization techniques using analogy with swarming principles and collective activities of social insects in nature have been used to the development of methodologies for solving a variety of real-world optimization problems. In this context, the social foraging behavior of Escherichia coli bacteria has recently been explored to develop a novel algorithm, called Bacterial Foraging Optimization Algorithm (BFOA). In this approach, the coordinates of a bacterium here represent an individual solution of the optimization problem, where the set of trial solutions converges towards the optimal solution following the foraging group dynamics of the bacteria population. The bacterium is free to move in the direction of positive nutrient gradient, i. e., increasing fitness. After a certain number of complete swims the best half of the population undergoes reproduction, eliminating the rest of the population. In order to escape local optima, an elimination-dispersion event is carried out so that some bacteria are eliminated at random with a very small probability and the new replacements are initialized at random locations of the search space. In this paper, the BFOA is applied to engineering systems, such as structural design and distillation column design. The results obtained are compared with those obtained from other classical evolutionary approaches.*

Keywords: *Bacterial Foraging, optimization, engineering system design.*

1. INTRODUCTION

Nowadays, the development of optimization techniques based on analogies with swarming principles and collective activities of social insects in nature (swarm intelligence) to solve real-world optimization problems constitutes an interesting and encouraging research topic. The corresponding techniques are based on various aspects of the collective activities of social insects, such as foraging of ants, birds flocking and fish schooling, which are self-organizing ones, meaning that complex group behavior emerges from the interactions of individuals who exhibit simple behaviors by themselves (Kennedy *et al*., 2001)

Swarm intelligence is inspired in nature, i.e., this characteristic found among living animals of a group contribute with their own experiences to the group, making it stronger in face of others. The most familiar representatives of swarm intelligence in optimization problems are: food-searching behavior of ants (Dorigo and Di Caro, 1999), particle swarm optimization (Shi and Eberhart, 2000), and artificial immune system (Castro and Timmis, 2002).

In this context, an optimization technique known as Bacterial Foraging Optimization Algorithm (BFOA) based on the foraging strategies of the *Escherichia coli* bacterium cells was recently proposed in (Passino, 2001). The basic idea is that during the course of evolution, bacteria colonies have developed sophisticated behavior, intricate communication capabilities, decentralized colony control, group foraging strategies and a high degree of worker cooperation when tackling tasks (Shapiro, 1988). Besides, natural selection tends to eliminate animals with poor foraging strategies through methods associated with locating, handling, and ingesting food and favors the propagation of genes of those animals that have successful foraging strategies, since they are more likely to obtain reproductive success. After many generations, poor foraging strategies are either eliminated or re-structured into good ones. Since a foraging organism/animal takes actions to maximize the energy utilized per unit time spent foraging, considering all the constraints presented by its own physiology, such as sensing and cognitive capabilities and environmental parameters (e.g., density of prey, risks from predators, and physical characteristics of the search area), natural evolution could lead to optimization. It is essentially this idea that could be applied to complex optimization problems (Farmer *et al.*, 1986; Passino, 2001; 2002).

In the literature, a small number of applications of BFOA are found, such as in engineering problems including antenna arrays (Mishra, 2005; Tripathy and Mishra, 2007; Guney and Basbug, 2008; Mangaraj *et al.*, 2008). Niu *et al.* (2006) has illustrated the faster settling time and higher robustness with BFA-PID controller. Kim *et al.* (2007) proposed a hybrid approach involving GA and BFA to tune PID controllers (proportional-integral-derivative controllers) of an automatic voltage regulator and has shown the efficiency of this approach for global optimization problems. Coelho and Sierakowski (2007) applied a new bacteria colony algorithm to solve the problem of path planning optimization of mobile robots.

In the present paper, BFOA is used in the context of engineering system design. This work is organized as follows. Section 2 provides a brief literature overview of the BFOA. The results and discussion are described in Section 3. Finally, the conclusions and suggestions for future work conclude the paper.

2. BACTERIAL FORAGING OPTIMIZATION ALGORITHM

2.1. Overview of chemotaxis behavior of *Escherichia coli*

Basically, chemotaxis is a foraging behavior that implements a type of optimization where bacteria try to climb up the nutrient concentration and avoid noxious substances and search for ways of neutral media (Coelho and Sierakowski, 2007). Based on these biological concepts, the definition of an optimization model of *Escherichia coli* bacterial foraging is possible.

In this sense, the algorithm proposed by Passino (2001, 2002) is based on the behavior of *Escherichia coli* present in human intestines. Its behavior and movement comes from a set of six rigid spinning (100–200 rotations per second) flagella, each driven as a biological motor. An *Escherichia coli* bacterium alternates through running and tumbling. Running speed is 10–20 μm/sec but they are unable to swim straight. The chemotaxis action of the bacteria is modeled as follows (Passino, 2001; 2002):

- In a neutral medium, if it tumbles and runs in an alternating fashion, its action could be similar to search.
- If swimming up a nutrient gradient (or out of noxious substances), or swimming for a longer period of time (climb up nutrient gradient or down noxious gradient), its behavior seeks increasingly favorable environments.
- If swimming down a nutrient gradient (or up noxious substance gradient), then the search action is avoiding unfavorable environments.

Subsequently, it can climb up nutrient hills and at the same time avoid noxious substances. The sensors it needs for optimal resolution are receptor proteins that are very sensitive and possess high gain. That is, a small change in the concentration of nutrients can cause a significant change in the behavior. Experts claim that this is probably the best-understood sensory and decision-making system in biology.

Mutations in *Escherichia coli* affect the reproductive efficiency at different temperatures, and occur at a rate of about 10-7 per gene per generation. *E. coli* occasionally engages in a conjugation that affects the characteristics of a population of bacteria. There are many types of taxis that are used in bacteria such as, aerotaxis (attracted to oxygen), phototaxis (light), thermotaxis (temperature), magnetotaxis (magnetic lines of flux) and some bacteria can change their shape and number of flagella based on the medium to reconfigure in order to ensure efficient foraging in a variety of media. Bacteria can form intricate stable spatio-temporal patterns in certain semisolid nutrient substances and they can radially eat their way through a medium if placed together initially at its center. Moreover, under certain conditions, they will secrete cell-to-cell attractant signals in order to group and protect each other.

2.2. Optimization function for the BFOA

BFOA is categorized into four processes: Chemotaxis, Swarming, Reproduction and Elimination. Each one is described as follows (Passino, 2001):

a) **Chemotaxis**: in this process the bacteria climbs the nutrient concentration, avoid noxious substances, and search for a way out of neutral media. The bacterium usually takes a tumble followed by a tumble or a tumble followed by a run. For an N_c number of chemotaxis steps the direction of movement after a tumble is given by:

$$\theta^i\left(j+1,k,l\right)=\theta^i\left(j,k,l\right)+C\left(i\right)\times\phi\left(j\right) \tag{1}$$

where $C(i)$ is the step size taken in direction of the tumble, j is the index for the chemotaxis step taken, k is the index for the number of reproduction step, l is the index for the number of elimination-dispersal event, $\phi\left(j\right)$ is the unit length random direction taken at each step.

If the cost at $\theta^i(j+1,k,l)$ is better than the cost at $\theta^i(j,k,l)$ then the bacterium takes another step of size $C(i)$ in that direction. This process will be continued until the number of steps taken is not greater than N_s.

b) **Swarming**: the bacteria in times of stresses release attractants to signal bacteria to swarm together. It also releases a repellant to signal others to be at a minimum distance from it. Thus all of them will have a cell to cell attraction via attractant and cell to cell repulsion via repellant. The equation involved in the process is:

$$J_{cc}\left(\theta,P\left(j,k,l\right)\right)=\sum_{i=1}^{S}J_{cc}^{i}\left(\theta,\theta^{i}\left(j,k,l\right)\right)$$
$$=\sum_{i=1}^{S}\left(-d_{attract}\exp\left(-w_{attract}\sum_{m=1}^{p}\left(\theta_{m}-\theta_{m}^{i}\right)^{2}\right)\right)+\sum_{i=1}^{S}\left(h_{repelent}\exp\left(-w_{repelent}\sum_{m=1}^{p}\left(\theta_{m}-\theta_{m}^{i}\right)^{2}\right)\right) \tag{2}$$

where $d_{attract}$ is the depth of the attractant, $w_{attract}$ is the measure of the width of the attractant, $h_{repelent}=d_{attract}$ is the height of the repellant effect, $w_{repelent}$ is the measure of the width of the repellant, p is the number of parameters to be optimized, S is the number of bacteria.

The bacteria climbing on the nutrient hill can be represented by:

$$J\left(i,j,k,l\right)+J_{cc}\left(\theta,P\right) \tag{3}$$

where $J(i,j,k,l)$ is the cost function.

c) **Reproduction**: after all the N_c chemotactic steps have been covered, a reproduction step takes place. The fitness (accumulated cost) of the bacteria is sorted in the ascending order. S_r ($S_r=S/2$) bacteria having higher fitness die and the remaining S_r are allowed to split into two, thus keeping the population size constant.

d) **Elimination-Dispersion**: for each elimination-dispersion event each bacterium is eliminated with a probability of p_{ed}. A low value of *Ned* dictates that the algorithm will not rely on random elimination-dispersal events to try to find favorable regions. A high value increases computational complexity but allows bacteria to find favorable regions. The p_{ed} should not be large either or else it should lead to an exhaustive search.

3. RESULTS AND DISCUSSION

For evaluating the methodology used in this work, some practical points should be emphasized:

- In all case studies the following parameters for the BFOA were used: population size (number of bacteria) equal to 32; number of chemotaxis steps equal to 10; length of a swim equal to 4; number of reproduction steps equal to 10; number of elimination-dispersion events equal to 2, probability that each bacteria will be eliminated/dispersed equal to 0.25, and penalization parameter equal to 10^8 (designed to penalize any constraint violation).
- In this paper, all case studies were run 30 times independently to obtain the values shown in the upcoming tables.

3.1. Welded beam design problem

The welded beam design problem is taken from Rao (1996) and He and Wang (2007), in which a welded beam is designed for minimum cost subject to constraints on shear stress (τ), bending stress in the beam, buckling load on the bar (P_c), end deflection of the beam (δ), and side constraints. There are four design variables as shown in Fig. 1, i. e., h (x_1), l (x_2), t (x_3), and b (x_4).

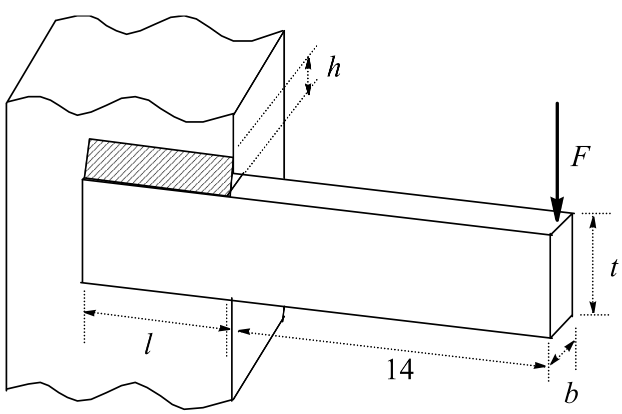

Figure 1: Welded beam design problem.

Mathematically, the problem can be formulated as follows (Rao, 1996):

$$\min\ f(x) = 1.10471 x_1^2 x_2 + 0.04811 x_3 x_4 (14 + x_2) \qquad (4)$$

subject to

$$g_1(x) = \tau(x) - 13000 \le 0 \qquad (5)$$

$$g_2(x) = \sigma(x) - 30000 \le 0 \qquad (6)$$

$$g_3(x) = x_1 - x_4 \le 0 \qquad (7)$$

$$g_4(x) = 1.10471 x_1^2 + 0.04811 x_3 x_4 (14 + x_2) - 5 \le 0 \qquad (8)$$

$$g_5(x) = 0.125 - x_1 \le 0 \qquad (9)$$

$$g_6(x) = \delta(x) - 0.25 \le 0 \qquad (10)$$

$$g_7(x) = 6000 - P_c(x) \le 0 \qquad (11)$$

where

$$\tau(x) = \sqrt{(\tau_1)^2 + 2\tau_1\tau_2 \frac{x_2}{2R} + (\tau_2)^2}, \qquad \tau_1 = \frac{6000}{\sqrt{2}x_1x_2}, \qquad \tau_2 = \frac{MR}{J}, \qquad M = 6000\left(14 + \frac{x_2}{2}\right), \qquad R = \sqrt{\frac{x_2^2}{4} + \left(\frac{x_1 + x_3}{2}\right)^2},$$

$$J = 2\left(\sqrt{2}x_1x_2\left(\frac{x_2^2}{12} + \left(\frac{x_1 + x_3}{2}\right)^2\right)\right), \ \sigma(x) = \frac{504000}{x_4 x_3^2}, \ \delta(x) = \frac{2.1952}{x_4 x_3^3}, \ P_c(x) = 64746.022(1 - 0.028234 x_3) x_3 x_4^3 .$$

The approaches applied to this problem include genetic algorithm with binary representation and traditional penalty function (Deb, 1991), a GA-based co-evolution model (Coello, 2000), and a co-evolutionary particle swarm optimization (He and Wang, 2007).

The following design space is adopted (He and Wang, 2007): 0.1 in $\le x_1 \le$ 2 in, 0.1 in $\le x_2 \le$ 10 in, 0.1 in $\le x_3 \le$ 10 in, 0.1 in $\le x_4 \le$ 2 in. The best solutions obtained by the above mentioned approaches are listed in Table 1. In this table, it can be seen that the best solution found by BFOA is better than the best solutions found by other techniques (Deb, 1991; Coello, 2000), but slightly inferior to the result obtained by He and Wang (2007).

Table 1. Comparison of the best solutions for the welded beam design problem using different techniques.

Design variables	Deb (1991)	Coello (2000)	He and Wang (2007)	This work (average)
x_1 (in)	0.248900	0.208800	0.202369	0.208796 (0.210056)
x_2 (in)	6.173000	3.420500	3.544214	3.412545 (3.435689)
x_3 (in)	8.178900	8.997500	9.048210	8.910044 (8.932545)
x_4 (in)	0.253300	0.210000	0.205723	0.210001 (0.215552)
g_1 (psi)	-5758.607	-0.337812	-12.839796	-23896.252
g_2 (psi)	-255.5769	-353.9026	-1.247467	-230.95874
g_3 (in)	-0.004400	-0.001200	-0.001498	-0.001204
g_4 ($)	-2.982866	-3.141865	-3.429347	-3.384378
g_5 (in)	-0.123900	-0.083800	-0.079381	-0.083796
g_6 (in)	-0.234160	-0.235649	-0.235536	-0.235222
g_7 (lb)	-44.65270	-363.2323	-11.681355	-808.56989
f ($)	2.433116	1.748309	1.728024	1.7318117 (1.745889)

3.2. Tension/compression string design problem

This problem is from Arora (1989), Belegundu (1982) and He and Wang (2007). It is devoted to the minimization of the weight of a tension/compression spring as shown in Fig. 2. The design variables are the wire diameter d (x_1), the mean coil diameter D (x_2) and the number of active coils P (x_3).

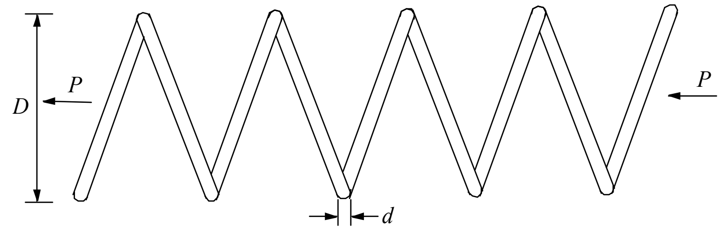

Figure 2: Tension/compression string design problem.

The mathematical formulation of this problem can be described as follows:

$$\min\ f(x) = (x_3 + 2)x_2 x_1^2 \tag{12}$$

subject to constraints on minimum deflection - Eq. (13), shear stress - Eq. (14), surge frequency - Eq. (15), limits on outside diameter - Eq. (16), and on side constraints:

$$g_1(x) = 1 - \frac{x_2^3 x_3}{71785 x_1^4} \le 0 \tag{13}$$

$$g_2(x) = \frac{4x_2^3 - x_1 x_2}{12566\left(x_2 x_1^3 - x_1^4\right)} + \frac{1}{5108 x_1^2} - 1 \le 0 \tag{14}$$

$$g_3(x) = 1 - \frac{140.45 x_1}{x_3 x_2^2} \le 0 \tag{15}$$

$$g_4(x) = \frac{x_1 + x_2}{1.5} - 1 \le 0 \tag{16}$$

The approaches applied to this problem include eight different numerical optimization techniques (Belegundu, 1982), a numerical optimization technique called constraint correction at constant cost (Arora, 1989), a GA-based co-evolution model (Coello, 2000), and a co-evolutionary particle swarm optimization (He and Wang, 2007).

The following design space is adopted (He and Wang, 2007): 0.05 in $\le x_1 \le$ 2 in, 0.25 in $\le x_2 \le$ 1.3 in, 2 in $\le x_3 \le$ 15 in. Table 2 presents the best solutions obtained by the above mentioned approaches. In this table, it can be seen that the best solution found by BFOA has the same quality of obtained by others techniques.

Table 2. Comparison of the best solutions for the tension/compression spring design problem using different methods.

Design variables	Belegundu (1982)	Arora (1989)	He and Wang (2007)	This work (average)
x_1 (in)	0.050000	0.053396	0.051728	0.051744 (0.050777)
x_2 (in)	0.315900	0.399180	0.357644	0.357754 (0.354189)
x_3 (in)	14.25000	9.185400	11.244543	11.56132 (11.70360)
g_1 (in)	-0.000014	0.000019	-0.000845	-0.028697
g_2 (ksi)	-0.003782	-0.000018	-1.260E-05	-0.000645
g_3 (-)	-3.938302	-4.123832	-4.051300	-3.911391
g_4 (in)	-0.727090	-0.698283	-0.727090	-0.727001
f (lb)	0.012674	0.012730	0.012674	0.012789 (0.012897)

3.3. Pressure vessel design problem

The pressure vessel design problem was proposed by Kannan and Kramer (1994) and is devoted to the minimization of the total cost of the specimen, including the cost of the material, forming and welding. A cylindrical vessel is capped at both ends by hemispherical heads as shown in Fig. 3. There are four design variables: T_s (x_1, thickness of the shell), T_h (x_2, thickness of the head), R (x_3, inner radius) and L (x_4, length of the cylindrical section of the vessel, not including the head). Among the four variables, T_s and T_h are integer multiples of 0.0625 in that are the available thicknesses of rolled steel plates, and R and L are continuous variables.

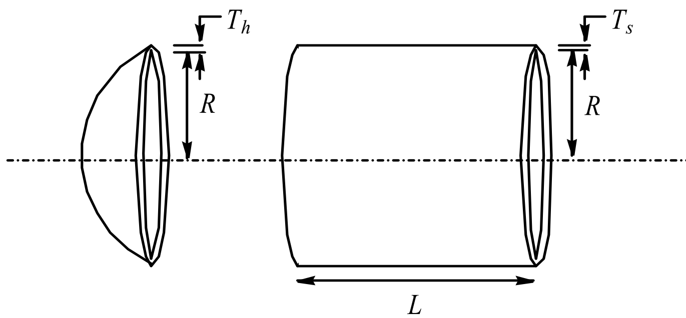

Figure 3: Pressure vessel design problem.

The problem can be formulated as follows (Kannan and Kramer, 1994):

$$\min f(x) = 0.6224x_1x_3x_4 + 1.7781x_2x_3^2 + 3.1661x_1^2x_4 + 19.84x_1^2x_3 \tag{17}$$

subject to

$$g_1(x) = -x_1 + 0.0193x_3 \leq 0 \tag{18}$$

$$g_2(x) = -x_2 + 0.00954x_3 \leq 0 \tag{19}$$

$$g_3(x) = -\pi x_3^2 x_4 - \frac{4}{3}\pi x_3^3 + 1296000 \leq 0 \tag{20}$$

$$g_4(x) = x_4 - 240 \leq 0 \tag{21}$$

In the literature, this problem has been solved by using an augmented Lagrangian multiplier approach (Kannan and Kramer, 1994), a genetic adaptive search (Deb, 1997), and a co-evolutionary particle swarm optimization (He and Wang, 2007).

In the present work, the following design space is adopted (He and Wang, 2007): 1 in $\leq x_1 \leq$ 99 in, 1 in $\leq x_2 \leq$ 99 in, 10 in $\leq x_3 \leq$ 200 in, 10 in $\leq x_4 \leq$ 200 in. The best solutions obtained by the above mentioned approaches are listed in Table 3. From Table 3, it can be seen that the best solution found by BFOA is better than the best solutions found by other techniques (Kannan and Kramer, 1994; Deb, 1997), and has same quality as the one obtained by He and Wang (2007).

Table 3. Comparison of the best solutions for the pressure vessel design problem using different methods.

Design variables	Kannan and Kramer (1994)	Deb (1997)	He and Wang (2007)	This work (average)
x_1 (in)	1.125000	0.937500	0.812500	0.812500 (0.875)
x_2 (in)	0.625000	0.500000	0.437500	0.437500 (0.500)
x_3 (in)	58.29100	48.32900	42.09126	42.09127 (42.12526)
x_4 (in)	43.69000	112.6790	176.7465	176.7466 (176.7747)
g_1 (in)	0.000016	-0.004750	-0.000139	-0.000139
g_2 (in)	-0.068904	-0.038941	-0.035949	-0.035949
g_3 (in^3)	-21.22010	-3652.876	-116.3827	-116.3827
g_4 (in)	-196.3100	-127.3210	-63.25350	-63.25350

f ($)	7198.0428	6410.3811	6061.0777	6061.0778 (6064.7256)

3.4. Binary distillation column design problem

Next case study is a binary distillation system from the MINOPT User's Guide (Schweiger *et al.*, 1997) and Bansal *et al.* (2003). The column has a fixed number of trays and the objective is to determine the optimal feed location (discrete decision), vapour boil-up, V, and reflux flow rate, R (continuous decisions), in order to minimize the integral square error (ISE) between the bottoms and distillate compositions and their respective set-points. The superstructure of the system is depicted in Fig. 4.

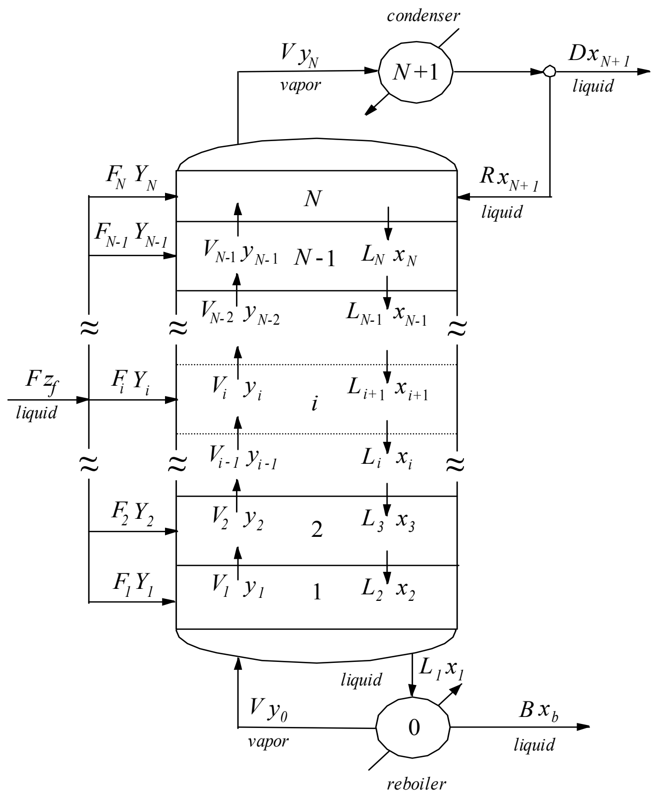

Figure 4: Binary distillation column.

The following modeling assumptions were used by Schweiger *et al.* (1997): (i) constant molar overflow; (ii) constant relative volatility, a; (iii) phase equilibrium; (iv) constant liquid hold-ups, equal to m for each tray and 10m for the reboiler and condenser; (v) no tray hydraulics; (vi) negligible vapour hold-ups; and (vii) no pressure drops. The system is initially at steady-state; at t=0 there is a step change in the feed composition, z_f, and the inequality constraints are that the distillate composition must be greater than 0.98, and the bottoms composition must be less than 0.02, at the end of the time horizon of 400 min. The problem can be stated mathematically as:

$$\min f\left(x\right)=ISE\left(t_{\mathrm{f}}\right) \tag{22}$$

where

$$\frac{d\left(ISE\right)}{dt}=\left(x_{\mathrm{b}}-x_{\mathrm{b}}^{*}\right)^{2}+\left(x_{\mathrm{N+1}}-x_{\mathrm{N+1}}^{*}\right)^{2} \tag{23}$$

subject to component balances - Eqs. (24)-(27), overall balances - Eq. (28)-(31), vapour-liquid equilibrium - Eqs. (32)-(33), and step disturbance – Eq. (24):

$$10m\frac{dx_{\mathrm{b}}}{dt}=L_{1}x_{1}-Vy_{\mathrm{o}}-Bx_{\mathrm{b}},\qquad \left.\frac{dx_{\mathrm{b}}}{dt}\right|_{t=0}=0 \tag{24}$$

$$m\frac{dx_{\mathrm{i}}}{dt}=L_{\mathrm{i+1}}x_{\mathrm{i+1}}-L_{\mathrm{i}}x_{\mathrm{i}}+V\left(y_{\mathrm{i-1}}-y_{\mathrm{i}}\right)+Fyf_iz_f,\quad \left.\frac{dx_{\mathrm{i}}}{dt}\right|_{t=0}=0,\quad \mathrm{i}=1,\ldots,\mathrm{N}\text{-}1 \tag{25}$$

$$m\frac{dx_{\mathrm{N}}}{dt}=-L_{\mathrm{N}}x_{\mathrm{N}}+V\left(y_{\mathrm{N-1}}-y_{\mathrm{N}}\right)+Fyf_Nz_f+Rx_{\mathrm{N+1}},\quad \left.\frac{dx_{\mathrm{N}}}{dt}\right|_{t=0}=0 \tag{26}$$

$$10m\frac{dx_{\mathrm{N+1}}}{dt}=V\left(y_{\mathrm{N}}-x_{\mathrm{N+1}}\right),\quad \left.\frac{dx_{\mathrm{N+1}}}{dt}\right|_{t=0}=0 \tag{27}$$

$$0=L_1-V-B \tag{28}$$

$$0=L_{\mathrm{i+1}}-L_{\mathrm{i}}+F\ yf_{\mathrm{i}},\quad \mathrm{i}=1,\ldots,\mathrm{N}\text{-}1 \tag{29}$$

$$0=-L_{\mathrm{N}}+V+F\ yf+R \tag{30}$$

$$0=V-D-R \tag{31}$$

$$y_{\mathrm{o}}=\frac{\alpha x_{\mathrm{b}}}{1+\left(\alpha-1\right)x_{\mathrm{b}}} \tag{32}$$

$$y_{\mathrm{i}}=\frac{\alpha x_{\mathrm{i}}}{1+\left(\alpha-1\right)x_{\mathrm{i}}},\quad \mathrm{i}=1,\ldots,\mathrm{N} \tag{33}$$

$$z_{\mathrm{f}}=0.54-0.09\exp\left(-10t\right) \tag{34}$$

The following design space is adopted (Bansal *et al.*, 2003): 0.05 kmol/min $\leq V \leq$ 2 kmol/min, 0.25 kmol/min $\leq R \leq$ 1.3 kmol/min, $1 \leq yf \leq 30$. Model parameters (Bansal *et al.*, 2003): number of trays (N) equal to 30; relative volatility, α equal to 2.5; tray liquid hold-up equal to 0.175 kmol; feed flow rate (F) equal to 1 kmol/min; distillate set-point ($x^*_{\mathrm{N+1}}$) equal to 0.98 and bottoms set-point (x^*_{b}) equal to 0.02.

Table 4 presents the results obtained by BFOA and by other techniques. In this table, it can be seen that the best solution found by BFOA has the same quality as compared with the results found by Bansal *et al.* (2003).

Table 4. Comparison of the best solutions for the binary distillation column design problem.

Design variables	Bansal *et al.* (2003)	This work (average)
Feed tray	25	25 (25)
V (kmol/min)	1.5426	1.5426 (1.5699)
R (kmol/min)	1.0024	1.0024 (1.0068)
f (-)	0.1817	0.181798 (0.183544)

4. CONCLUSIONS

In this work, Bacterial Foraging Optimization Algorithm (BFOA) based on the social foraging behavior of the *Escherichia coli* bacteria, was applied to solve different design problems. The simulation results were compared with those obtained from other competing evolutionary algorithms. Besides, the results showed that the methodology is configured as a promising alternative for a number of engineering applications. However, in terms of the number of objective function evaluations, this approach need yet to be better studied, so that more definitive conclusions can be drawn. This particular characteristic, i.e., the number of objective function evaluations is inherent to this methodology due to the quantity loops to be performed. Consequently, it is expected that a high number of objective function evaluations is necessary in the present version of the algorithm.

Further research work will be focused on the influence of the parameter values required by BFOA on the quality of solution.

5. ACKNOWLEDGEMENTS

Dr. Lobato acknowledges the financial support provided by FAPEMIG Proc. No. 00233/08. Dr. Steffen acknowledges the financial support provided by CNPq (INCT-EIE).

6. REFERENCES

Arora, J. S., 1989. “Introduction to Optimum Design”. McGraw-Hill, New York.
Bansal, V., Sakizlis, V., Ross, R., Perkins, J. D., Pistikopoulos, E. N., 2003, “New Algorithms for Mixed-integer Dynamic Optimization”, Computers and Chemical Engineering, 27, 647-668.
Belegundu, A. D., 1982. “A Study of Mathematical Programming Methods for Structural Optimization”. Department of Civil and Environmental Engineering, University of Iowa, Iowa City, Iowa.
Castro, L. N., and Timmis, J. I., 2002, Artificial Immune Systems: A New Computational Intelligence Approach, Springer-Verlag, London, UK.
Coello, C. A. C., 2000. “Use of a Self-adaptive Penalty Approach for Engineering Optimization Problems”. Computers in Industry 41, 113–127.
Coelho, L. S. and Sierakowski, C. A., 2007, “Bacteria Colony Approaches with Variable Velocity Applied to Path Optmization of Mobile Robots”, 19th International Congress of Mechanical Engineering – COBEM 2007. Brasília, Brazil.
Deb, K., 1991. “Optimal Design of a Welded Beam via Genetic Algorithms”. AIAA Journal, 29 (11), 2013–2015.
Deb, K., 1997. “GeneAS: A Robust Optimal Design Technique for Mechanical Component Design”. In: Dasgupta, D., Michalewicz, Z. (Eds.), Evolutionary Algorithms in Engineering Applications. Springer, Berlin, pp. 497–514.
Dorigo, M., and Di Caro, G., 1999, “The Ant Colony Optimization Meta-Heuristic”, New Ideas in Optimization, Edited by D. Corne, M. Dorigo, and F. Glover, McGraw-Hill, pp. 11-32.
Farmer, J. D., Packard, N. H., Perelson, A. S., 1986, “The Immune System, Adaptation, and Machine Learning”, Physica D 22, 187–204.
Guney, K. and Basbug, S., 2008, “Interference Suppression of Linear Antenna Arrays by Amplitude-only Control using a Bacterial Foraging Algorithm”, Progress In Electromagnetics Research, PIER 79, 475–497.
He, Q. and Wang, L., 2007, “An Effective Co-evolutionary Particle Swarm Optimization for Constrained Engineering Design Problems”, Engineering Applications of Artificial Intelligence, 20, 89-99.
Kannan, B. K., Kramer, S. N., 1994. “An Augmented Lagrange Multiplier based Method for Mixed Integer Discrete Continuous Optimization and its Applications to Mechanical Design”. Transactions of the ASME, Journal of Mechanical Design 116, 318–320.
Kennedy, J. F., Eberhart, R. C., and Shi, R. C., 2001, Swarm Intelligence, San Francisco: Morgan Kaufmann Pub.
Kim, D. H., Abraham, A. and Cho, J. H., 2007, “A Hybrid Genetic Algorithm and Bacterial Foraging Approach for Global Optimization”, Information Sciences, 177, 3918–3937.
Mangaraj, B. B., Misra, I. S. and Barisal, A. K., 2008, “Optimizing Included Angle of Symmetrical V-dipole for Higher Directivity using Bacteria Foraging Algorithm”, Progress In Electromagnetic Research B, 3, 295–314.
Mishra, S., 2005, “A Hybrid Least Square-fuzzy Bacteria Foraging Strategy for Harmonic Estimation”, IEEE Trans. Evol. Comput., 9 (1), 61–73.
Niu, B., Zhu, Y., He, X. and Zeng, X., 2006, “Optimum Design of PID Controllers using only a Germ of Intelligence”, 6th World Congress on Intelligent Control and Automation, 3584–3588, Dalian, China.
Passino, K. M., 2001, “Biomimicry of Bacterial foraging for Distributed Optimization, University Press, Princeton, New Jersey.
Passino, K. M., 2002, “Biomimicry of Bacterial Foraging for Distributed Optimization and Control”, Control Systems Magazine, IEEE, 22 (3), 52-67.
Rao, S. S., 1996. “Engineering Optimization”. Wiley, New York.
Schweiger, C. A., Rojnuckarin, A., Floudas, C. A., 1997. “MINOPT: User’s Guide”. Princeton University. Software Version 2.0.
Shapiro, J. A., 1988, “Bacteria as Multicellular Organisms”, Scientific American, Vol. 256, No. 6, pp. 62-69.
Shi, Y. and Eberhart, R. C., 2000, “Experimental Study of Particle Swarm Optimization”, Proceedings of 4th World Conference on Systems, Cybernetics and Informatics, Orlando, FL, USA.
Tripathy, M. and Mishra, S., 2007, “Bacteria Foraging-based Solution to Optimize both Real Power Loss and Voltage Stability Limit”, IEEE Transactions on Power Systems, 22 (1), 1-11.