SYM-OP-IS 2014
XLI Simpozijum o operacionim istraživanjima
Divčibare, 16-19. 09. 2014.
Univerzitet u Beogradu
Saobraćajni fakultet

ANALIZA KONVERGENCIJE PO MODELU KONSTRUKTIVNE VERZIJE ALGORITMA OPTIMIZACIJE KOLONIJOM PČELA

MODEL CONVERGENCE PROPERTIES OF THE CONSTRUCTIVE BEE COLONY OPTIMIZATION ALGORITHM

TATJANA JAKŠIĆ KRÜGER[1], TATJANA DAVIDOVIĆ[1]

[1] Matematički institut SANU, Beograd, {tatjana, tanjad}@mi.sanu.ac.rs

---

Rezime: *Metoda optimizacije kolonijom pčela* (*eng. Bee Colony Optimization, BCO*) *je metaheuristička metoda, inspirisana prirodnim procesima, pomoću koje se rešavaju teški optimizacioni problemi. BCO metoda je zasnovana na ponašanju pčela u prirodi tokom potrage za hranom i predložili su je Lučić i Teodorović* 2001. *godine. Kroz brojne primene u transportu, teoriji lokacija, raspoređivanju i drugim oblastima metoda je evoluirala, odnosno prošla kroz mnoge transformacije, modifikacije, pa i paralelizacije, što je uticalo na povećanje njene efikasnosti. Nedavno su se pojavili prvi rezultati na temu teorijske verifikacije BCO metode, a ovaj rad predstavlja korak dalje u tom smeru.*

Ključne reči: *Operaciona istraživanja, prirodom inspirisani algoritmi, inteligencija roja, potraga pčela za nektarom, kombinatorna optimizacija, metaheurističke metode, uslovi konvergencije.*

Abstract: *The Bee Colony Optimization* (*BCO*) *algorithm is a nature-inspired meta-heuristic method for dealing with hard, real-life optimization problems. It is based on the foraging habits of honeybees and was proposed by Lučić and Teodorović in* 2001. *Through numerous applications in transportation, location theory, scheduling and some other fields, method has evolved and underwent many transformations, modifications, even parallelization, which resulted in the increase of its efficiency. The first results of the theoretical verification of the BCO method have appeared recently. The aim of this paper is to further contribute to this topic.*

Keywords: *Operational research, bio-inspired algorithms, swarm intelligence, foraging of honeybees, combinatorial optimization, meta-heuristic methods, convergence properties.*

1. INTRODUCTION

BCO is a population-based meta-heuristic method that belongs to the class of Swarm intelligence algorithms. Lučić and Teodorović were among the first who used the basic principles of collective bee intelligence in dealing with combinatorial optimization problems (Lučić and Teodorović 2001). The basic idea behind BCO is to build the multi agent system (colony of B artificial bees) that will search for good solutions of various combinatorial optimization problems exploring the principles used by honey bees during nectar collection process. Every artificial bee generates one solution to the problem. In order to find the best possible solutions, autonomous artificial bees collaborate and exchange information. Using collective knowledge and information sharing, artificial bees concentrate on the more promising areas, and slowly abandon solutions from the less promising ones. Step by step, artificial bees collectively generate (Davidović *et al.* 2012) and/or improve their solutions (Davidović *et al.* 2011). The BCO search is running in iterations until some predefined stopping criterion is satisfied. An iteration consists of predefined number (NC) of steps. Each algorithm step consists of two alternating phases: *forward pass* and *backward pass*. During each forward pass, every bee is exploring the search space. Backward pass is responsible for the collaboration (knowledge exchange) among bees. More detailed description of the BCO method can be found in (Davidović *et al* 2014) and its applications are reviewed in (Teodorović *et al.* 2014).

Our main objective is to contribute to the mathematical verification of the BCO meta-heuristic, aiming to reduce the gap between successful practice and missing theory. Like other meta-heuristics, BCO method suffers from a serious drawback related to the actual quality of the reported solution: even if this solution is optimal there is no any proof. For this reason, in the recent literature a lot of effort has been done on the theoretical analysis of meta-heuristic methods: (Brimberg *et al.* 2004) for Variable Neighbourhood Search

VNS, (Granville *et al.* 1994), (Hajek 1988) and (Steinhöfel *et al.* 2000) for Simulated Annealing, SA, (Gutjahr 2002) and (Stützle and Dorigo 2002) for Ant Colony Optimization ACO, (Hanafi 2001) for Tabu Search TS, (Hartl 1990), (Rudolph 1994) and (Schmitt 2001) for Genetic Algorithm GA, (Jacobson and Yücesan 2004) for Generalized Hill Climbing Algorithms GHC, (Jiang *et al.* 2007) and (Zeng *et al.* 2004) for Particle Swarm Optimization PSO, (Margolin 2005) for Cross Entropy.

In the recent papers (Jakšić Kruger 2013) and (Jakšić Kruger *et al.* 2014) two types of convergence of the BCO method are analyzed: the so called *best-so-far* convergence and the more sophisticated *model* convergence. Both convergence analyses are considered for constructive variant of the BCO algorithm where the complete solution contains subset of components. It was noted that the implementation of the forward pass is crucial for the convergence, while the backward pass can influence the convergence speed.

The rest of the paper is organized as follows. The next section provides a basic notation and definitions of convergence for meta-heuristic methods. In Section 3 we present mathematical validation and proof for model convergence of constructive BCO. Section 4 concludes the paper.

2. BASIC NOTATION AND DEFINITIONS

The term *convergence* is defined in order to answer the question whether or not the current solutions proposed by a considered meta-heuristic method converge to an optimal solution and if yes, how fast this happens (Gutjahr 2009). The aforementioned paper provides the basic ideas of convergence proofs for various meta-heuristic methods. We use the same notation and definitions, and recall them briefly here.

Let us denote by $(x_1,x_2,\ldots x_t, \ldots)$, $x \in \chi$ the sequence of elements that are under consideration. We also define some distance function d between these elements. The sequence (x_t) converges to a limit x^*, if for each $\varepsilon > 0$, there is an integer t_0 such that $d(x_t, x^*) < \varepsilon$ for all $t \geq t_0$. If the space χ is finite, the simplified version of the definition is applicable: x_t converges to x^* if and only if there is some t_0 such that $x_t = x^*$ for all $t \geq t_0$. When meta-heuristics are under consideration, the analyzed sequence $(x_1,x_2,\ldots x_t, \ldots)$, with $x_t \in \chi$, denotes the sequence of "best-so-far" solutions x_t^{bsf} provided by a given meta-heuristic method after t iterations. The distance function d is defined as $d(x_i,x_j) = |f(x_i)-f(x_j)|$, i.e., as the absolute difference between the corresponding values of the objective function. For most of the meta-heuristic methods it holds that once an optimal solution is reached, it is propagated through the forthcoming iterations. Consecutively, our sequence becomes $(x_1^{bsf}, x_2^{bsf}, \ldots, x_t^{bsf}, \ldots, x^*, x^*, \ldots)$.

Most of the meta-heuristics are stochastic search algorithms, and therefore, in order to obtain more formal definition of convergence we need to consider the series of random variables with a common distribution. In general, the random variables are not independent, and when meta-heuristics are considered the independency holds only for random walk search techniques. The two important established definitions of stochastic convergence are as follows.

Definition 1: A sequence of random variables $(X_1, X_2, \ldots)$ converges *with probability one* (short: w. pr. 1) or *almost surely* to a random variable X^*, if $Pr\{\lim_{t \to \infty} X_t = X^*\} = 1$, i.e., if with probability one, the realization $(x_1,x_2,\ldots)$ of the sequence X_t converges to the realization x^* of X^*.

Definition 2: A sequence of random variables $(X_1, X_2,\ldots)$ converges *in probability* to a random variable X^*, if for all $\varepsilon > 0$, $Pr\{ d(X_t, X^*) \geq \varepsilon\} \to 0$ as $t \to \infty$, where d is the distance function on the space χ from which the random variables X_t take their values.

If χ is a finite set, convergence of X_t to x^* w. pr. 1 holds exactly if $Pr\{\exists u \geq 1: X_t = x^*, \forall t \geq u\} = 1$, and convergence of X_t in probability holds exactly if $Pr\{X_t = x^*\} \to 1$ as $t \to \infty$.

Zlochin et al. (2004) and Gutjahr (2009) discussed two main approaches to constructing meta-heuristic methods. In Zlochin (2004) search methods were defined as *instance-based* type if new candidate solutions are being generated using solely the current solution or the current population of solutions. Gutjahr (2009) classified these methods as convergent according to the *best-so-far* criterion since only the best-so-far solution is being considered, while other parameters are not analyzed. Best-so-far convergence is easy to prove but it is not of practical value.

On the other hand, in both of the above mentioned papers the alternative frameworks, that enable improving the search, were proposed. These frameworks are based on analyzing the parameters of the corresponding meta-heuristic method. The authors agree that, in order to generate high-quality solutions, the considered meta-heuristic has to learn from previously visited solutions how to concentrate its search on the regions containing solutions of higher quality. According to (Zlochin *et al.* 2004), a meta-heuristic method satisfies the *model-based search properties* if it attempts to solve the optimization problem by repeating the following two steps:

- Candidate solutions are constructed using some parameterized probabilistic model,

- Candidate solutions are used to modify the model in such a way to concentrate the search toward more promising regions (containing solutions of better quality).

For that type of meta-heuristics the *model-based parameter scheme* was adapted as an assurance for model convergence by (Gutjahr 2009). Then, the considered meta-heuristic is being analyzed with respect to its parameters. The main conclusion in (Gutjahr 2009) and (Zlochin *et al.* 2004) is that parameter values have to change during the search.

In the model-based view, the generation of new search points is depending on the *current model* (the current set of solutions), (Gutjahr 2009). The newly obtained cost function values are evaluated, and the obtained information is used to modify the model. According to (Gutjahr 2009) the *model convergence* is described by requirement that the model converge as $t \to \infty$, to some state that supports only the generation of optimal or at least high quality solutions.

Contrary to the proofs of best-so-far convergence which are quite easy, model convergence proofs have to take the exploration/exploitation tradeoff explicitly into account and only succeed under parameter assumptions ensuring a proper balance between these two factors. Typically, the convergence results lead to rather narrow conditions for parameter schemes within which model convergence holds; outside the balanced regime, either a surplus of exploitation yields premature convergence to a suboptimal solution, or a surplus of exploration produces random-search-type behavior without model convergence (although best-so-far convergence may hold).

In analyzing the parametric properties of a meta-heuristic methods according to (Gutjahr 2009), and (Jacobson and Yücesan 2004) the following terms are important. For each iteration t it holds:

- $C(t)$ represents the event that $x_t \in X^*$, i.e., an optimal solution was generated in iteration t. The complementary event is denoted by $C^c(t)$.
- $B(t)$ denotes the event $C^C(1) \cap C^C(2) \cap ... \cap C^C(t)$, i.e., the algorithm does not visit any element of X^* over first t iterations. $B^c(t)$ is the complementary event to $B(t)$.
- $B = \cap_{t=1}^{\infty} B(t)$ describes the event that an optimal solution cannot be generated by the algorithm, i.e., no iteration at all produces an optimal solution.
- $r(t) = P\{B^C(t) \mid B(t\text{-}1)\} = P\{C(t) \mid B(t\text{-}1)\}$, is the probability that in the iteration t, an optimal solution is produced for the first time.

According to Definition 2, the convergence of x_t in probability to the set X^* can be expressed as $P_r\{C(t)\} \to 1$ as $t \to \infty$. We now reproduce the relevant main theorem for the convergence of GHC algorithm and an auxiliary result formulated as Lemma 1.

Theorem 1*: A GHC algorithm converges in probability to X* if and only if the following two conditions are satisfied:*

$$(i)\sum_{t=1}^{+\infty} r_t = +\infty$$

$$(ii)\, P_r\{C^c(t) \mid B^c(t-1)\} \to 0 \text{ as } t \to \infty.$$

Lemma1*:* $P_r(B) = 0 \Leftrightarrow \sum_{t=1}^{+\infty} r_t = +\infty.$

The detailed proofs can be found in (Jacobson and Yücesan 2004) and (Gutjahr 2009).

3. MODEL CONVERGENCE OF CONSTRUCTIVE BCO

In (Jakšić Kruger *et al.* 2014) the authors analyzed both types (best-so-far and model) of convergence properties of the BCO method. They established the conditions that are sufficient for the model convergence of the constructive variant of the BCO algorithm, in the case where not all the components are included in the solution. Here we are concerned with proving model convergence of the constructive BCO algorithm when all the components are included in the solution. This scenario occurs while dealing with traveling salesman problem TSP, vehicle routing problems, or scheduling problems.

The (model convergent) variants of the BCO algorithm should always represent a highly structured search procedure which exploits the historical record of performance reflected at each stage of its execution. These requirements could be fulfilled if the forward pass includes some learning properties. As in our previous paper, in order to assure the generation of high quality solutions, we should change the selection probability for the components based on previously obtained good solutions. In fact, our selection probability depends on two factors: the problem specific and the learned one.

Let us consider first TSP problem and denote by (i,j) a pair of components that are directly connected. We propose the following modification scheme for the selection probability of component j in the iteration t after we chose component i:

$$p_{(i,j)}(t+1) = \begin{cases} 1-\lambda_t(1-p_{(i,j)}(t)) \text{ if } (i,j) \in x_t^{bsf}, \\ \lambda_t\, p_{(i,j)}(t) \text{ if } (i,j) \notin x_{t-1}^{bsf} \end{cases} \tag{1}$$

where λ_t represents the time dependent *learning rate*. The idea is to learn from the previous experience how each component influences the quality of generated solution. If the pair of components (i,j) was a part of the current best (best-so-far) solution, the probability that it will be selected in the next iteration is increased. If this pair of components is included in some low quality solutions, we decrease the probability of its selection for the next iteration. Now we can present the sufficient conditions that the BCO algorithm, with the above defined selection probability modification scheme, converges in probability to an optimal solution.

Theorem 2: *Assume that*

$$1 \geq \lambda_t \geq \frac{\log t}{\log(t+1)} \textit{ for all } t \geq t_0 \textit{ with } t_0 \geq 1 \tag{2}$$

and

$$\sum_{t=1}^{+\infty}(1-\lambda_t) = +\infty. \tag{3}$$

Then the corresponding BCO algorithm converges in probability to one of the optimal solutions.

Proof: We have to proof that the conditions (i) and (ii) from Theorem 1 are satisfied.

(i) We will actually prove the equivalent condition (according to Lemma 1), i.e., that $Pr(B)=0$. As it is already defined, $C(t)$ denotes the event that iteration t is the first in which an optimal solution is found by some bee. Consider a fixed optimal solution x^*. Then it holds:

$$B = C^C(1) \cap C^C(2) \cap \ldots \Rightarrow x^* \text{ is never found and hence,}$$

$$\begin{aligned} Pr(B) &= Pr\{C^c(1) \cap C^c(2) \cap \ldots\} \\ &\leq Pr\{x^* \text{ is never found}\} \\ &= \prod_{t=1}^{+\infty} \Pr\{x^* \textit{ is not found in iteration } t \mid x^* \textit{ is not found in iteration } k<t\}. \end{aligned} \tag{4}$$

According to the cumulative probability update rule (1), for all pairs of components (i,j) that don't belong to x^{bsf} it holds:

$$p_{(i,j)}(t) = \left[\prod_{k=1}^{t}\lambda_k\right] \cdot p_{(i,j)}(0),$$

which can be easily verified by induction. From the condition (2) it holds

$$\left[\prod_{k=1}^{t}\lambda_k\right] \cdot p_{(i,j)}(0) \geq \left[\prod_{k=1}^{t_0-1}\lambda_k\right] \cdot \left[\prod_{j=t_0}^{t}\frac{\log j}{\log(j+1)}\right] \cdot p_{(i,j)}(0) = \left[\prod_{k=1}^{t_0-1}\lambda_k\right] \cdot \frac{\log t_0}{\log t}\, p_{(i,j)}(0) = \frac{const}{\log t}.$$

Therefore, the above derived is a lower bound of the worst case selection scenario for any component (i,j). Consecutively, for the probability to find the optimal solution x^* by any bee it holds:

$$P^*(t) = \prod_{(i,j)\in x^*} p_{(i,j)}(t) \geq \left(\frac{const}{\log t}\right)^n,$$

where n denotes the total number of components (number of cities in the TSP).

Considering the complementary event (the optimal solution x^* was not found by any bee) we obtain the upper bound on the right hand side of the relation (4) as:

$$\prod_{t=t_0}^{+\infty}\left[1-\left(\frac{const}{\log t}\right)^n\right].$$

Applying a logarithm to this expression gives us:

$$\sum_{t=t_0}^{+\infty}\left[1-\left(\frac{const}{\log t}\right)^n\right] \le -\sum_{t=t_0}^{+\infty}\left(\frac{const}{\log t}\right)^n = -\infty.$$

From this we can conclude

$$\prod_{t=t_0}^{+\infty}\left[1-\left(\frac{const}{\log t}\right)^n\right] = 0,$$

i.e., in (4) we have $Pr(B) \le 0$. Since $Pr(B) \ge 0$ always holds, we have $Pr(B) = 0$.

(*ii*) This condition actually means the following: once the optimal solution is found the probability that it will not be generated in the next iteration tends to zero, as the number of iterations tends to infinity.

Let m denote the index of the iteration when x^* is generated for the first time. Then in all iteration $t > m$, the selection probability for components not included in the optimal solution converge to zero as the number of iterations tends to infinity.

Consider the pair of components $(i,j) \notin x^*$. According to (1), its selection probability after iteration m, i.e., in some iteration $m+r$, $r = 1, 2,...$ is modified as follows:

$$p_{(i,j)}(m+r) = \left[\prod_{k=m+1}^{m+r}\lambda_k\right]\cdot p_{(i,j)}(m).$$

Due to the condition (3) we have

$$\sum_{t=1}^{+\infty}(1-\lambda_t) = +\infty, \text{ i.e., } \prod_{k=1}^{+\infty}\lambda_k = 0.$$

Therefore, after generating the optimal solution x^*, for the probability that the pair of component $(i,j) \notin x^*$ will be again used it holds:

$$\lim_{t\to+\infty} p_{(i,j)}(t) = \lim_{t\to+\infty}\left[\prod_{k=m+1}^{t}\lambda_k\right]\cdot p_{(i,j)}(m) = 0,$$

which competes the proof of the theorem. ■

Considering the problem of scheduling independent tasks on identical processors, where the pair of components (i,j) describes the situation that task j is allocated to processor i, we can apply the selection probability modification scheme defined by (1). Therefore, the above described reasoning holds, and the resulting BCO algorithm satisfies model convergence properties.

5. CONCLUSION

We analyzed some convergence properties of constructive variant of the Bee Colony Optimization method (BCO) in cases when solutions contain all the components. We established the conditions that are sufficient for the model convergence of the constructive BCO and provided directions for designing the corresponding optimization algorithm. The possible topic of future research may include expanding the existing convergence results to improvement variant of BCO as well as performing the analysis of the convergence speed.

BIBLIOGRAPHY

[1] Brimberg, J., Hansen, P., & Mladenović, N. (2004). Convergence of variable neighborhood search. Groupe d'études et de recherche en analyse des décisions, HEC Montréal.

[2] Davidović, T., Teodorović, D., Šelmić, M. (2014). Bee Colony Optimization Part I: The Algorithm Overview. YUJOR, (accepted).

[3] Davidović, T., Šelmić, M., Teodorović, D., & Ramljak, D. (2012). Bee colony optimization for scheduling independent tasks to identical processors. J. Heur., 18(4), 549-569.

[4] Davidović, T., Ramljak, D., Šelmić, M., & Teodorović, D. (2011). Bee colony optimization for the p-center problem. Comput. Oper. Res., 38(10), 1367-1376.

[5] Granville, V., Kriváanek, M., & Rasson, J-P. (1994). Simulated annealing: A proof of convergence. IEEE Pattern Anal., 16(6), 652–656.

[6] Gutjahr, W. J. (2002). ACO algorithms with guaranteed convergence to the optimal solution. Inform. Process. Lett., 82(3), 145-153.

[7] Gutjahr, W.J. (2009). Convergence analysis of metaheuristics. In V. Maniezzo, T. Stützle, & S. Voss (Eds.), Matheuristics: hybridizing metaheuristics and mathematical programming (159–187). Springer.

[8] Hajek, B. (1988). Cooling schedules for optimal annealing. Math. Oper. Res., 13(2), 311-329.

[9] Hanafi, S. (2001). On the convergence of tabu search. J. Heur., 7(1), 47-58.

[10] Hartl, R. F. (1990). A global convergence proof for a class of genetic algorithms. Technical report, University of Technology, Vienna.

[11] Jacobson, S. H., & Yücesan, E. (2004). Global optimization performance measures for generalized hill climbing algorithms. J. Global Optim., 29(2), 173-190.

[12] Jakšić Krüger, T. (2013). On the convergence of the bee colony optimization meta-heuristic method (in Serbian). 4th Symp. Math. Applications, Belgrade, Serbia, May, 24-25, 2013.

[13] Jakšić Krüger, T., Davidović T., Teodorović, D., Šelmić, M. (2014). The BCO algorithm and its Convergence. Int. J. Bio-inspired Computing (accepted).

[14] Jiang, M., Luo, Y. P., & Yang, S.Y. (2007). Stochastic convergence analysis and parameter selection of the standard particle swarm optimization algorithm. Inform. Process. Lett., 102(1), 8-16.

[15] Lučić P. & Teodorović D. (2001). Bee system: modeling combinatorial optimization transportation engineering problems by swarm intelligence. In: Preprints of the TRISTAN IV Triennial Symposium on Transportation Analysis (441–445), Sao Miguel, Azores Islands, Portugal.

[16] Margolin, L. (2005). On the convergence of the cross-entropy method. Ann. Oper. Res., 134(1), 201-214.

[17] Rudolph, G. (1994). Convergence analysis of canonical genetic algorithms. IEEE T. Neural Networ., 5(1), 96-101.

[18] Schmitt, L. M. (2001). Theory of genetic algorithms. Theor. Comput. Sci., 259, 1-61.

[19] Steinhöfel, K., Albrecht, A., and Wong, C.-K. (2000).Convergence analysis of simulated annealing-based algorithms solving flow shop scheduling problems. In Algorithms and Complexity (277-290), Springer.

[20] Stützle, T. & Dorigo, M. (2002). A short convergence proof for a class of ant colony optimization algorithms. IEEE T. Evolut. Comput., 6(4), 358-365.

[21] Teodorović, D., Šelmić, M., Davidović, T., Bee Colony Optimization Part II: The Applications Survey, YUJOR, (accepted 2014).

[22] Zeng, J-C., & Cui, Z-H. (2004). A guaranteed global convergence particle swarm optimizer. J. Comput. Res. Develop., 41(8), 1333-1338.

[23] Zlochin, M., Birattari, M., Meuleau, N., & Dorigo, M. (2004). Model-based search for combinatorial optimization: A critical survey. Ann. Oper. Res., 131(1–4), 373–395.