World Applied Sciences Journal 22 (12): 1789-1797, 2013
ISSN 1818-4952
© IDOSI Publications, 2013
DOI: 10.5829/idosi.wasj.2013.22.12.13022

Bioinspired Algorithm Applied to Solve the Travelling Salesman Problem

V.V. Kureichik and Yu. A. Kravchenko

Southern Federal University, Taganrog, Russia

Abstract: The article is devoted to the study and development of modified bioinspired algorithm and the experimental studies of its characteristics to solve the travelling salesman problem. This algorithm is part of the swarm intelligence method, which is one of bioinspired approaches describing the collective behavior of a decentralized self-organizing system and consists of a multitude of agents (ants), locally interacting between each other and the environment. Ants belong to social insects living inside of a collective - the colony. The self-organization includes the numerous mechanisms ensuring the achievement of the global target by a system as a result of insignificant interaction of elements in this system and is the basis of the "social" behavior of ants. Implementation of the local information by system's elements is the principal feature of this interaction. This eliminates any centralized control and appeal to the global image representing the system in the external environment. Self-organization is a result of interaction between four components: randomness, multiplicity, positive and negative feedbacks. The computational experiment was carried out. The series of tests and experiments have specified the theoretical estimates of the time complexity of the proposed modified bioinspired algorithm. At the best, the time complexity of the algorithm ≈ $O(nlogn)$ and in the worst case – $o(n^3)$.

Key words: Ant's Colony · Adaptation · Travelling Salesman Problem · Bioinspired Algorithm · Swarm Intelligence

INTRODUCTION

Ant's algorithms are the probabilistic greedy heuristics, where the probabilities are fixed and based on information about the quality of solutions obtained from previous decisions. These can be used to solve both static and dynamic combinatorial optimization problems. The convergence is guaranteed, i.e. the optimal solution will be obtained in any case, however, the rate of convergence is unknown [1].

The concept of ??the ant's algorithm is the modeling of ant's behavior related to their ability to quick finding of the shortest path from the nest to a food source and adapt to the changing conditions by finding of a new shortest path. By moving, an ant marks the path by pheromone and this information is used by other ants to select the path. This elementary rule of behavior determines the ability of ants to find a new path if the previous is unavailable.

Since the modeling of the movements of ants in some directions is the basis of ant's algorithm, then this approach can be an effective method to find rational solutions for optimization problems, admitting a graph interpretation. A series of experiments show that the efficiency of ant's algorithms increases with the dimension of solution of the optimization problems. Significant results can be obtained for nonstationary systems with time-varying parameters.

Ant's algorithms in comparison with genetic algorithms (GAs) have some advantages because these are:

- Based on the memory of the entire colony instead of the memory only about the previous generation;
- Less susceptible to non-optimal initial decisions (due to the random choice of the path and the memory of the colony) [2].

The disadvantages include the following properties of ant's algorithms:

- Theoretical analysis is difficult (as a result of a sequence of random decisions; the distribution of probable changes in iterations; research is more experimental rather than theoretical);

Corresponding Author: Kravchenko, Southern Federal University, Nekrasovskiy lane. 44, Taganrog, 347928 Russia.

- Convergence is guaranteed, but the convergence time is not defined;
- The implementation of additional methods such as local search is required;
- Strongly depend on the adjustment parameters which are selected on the basis of experiments.

Non-convergence is an important feature of ant's algorithms because numerous variants of solutions are simultaneously studying after a large number of iterations what results in long time delays in the local extremes. On-line adaptation of the parameters using the fuzzy rule base and their hybridization with other methods of natural computing such as genetic algorithms are the prospective methods to improve the ant's algorithms. Hybridization can occur on the island circuit when various algorithms solve the problem independently and autonomous (each on a separate "island") with the exchange by the best solutions after a certain time or by a "master-apprentice" principle when the general algorithm is "master" and conveys the decision of typical subtasks to "apprentice" – specialized and fast algorithm.

The aim of this study was to develop the modified bioinspired algorithm and investigate its characteristics to solve the traveling salesman problem.

The traveling salesman problem is a task which requires to find the Hamiltonian cycle of minimum total weight (minimum path passed by the agents) in complete disoriented final graph.

The general idea of bioinspired algorithm is the elucidation of the shortest path passed by almost blind agents, like ants from their colony to feeding sources.

According to the study results, we provide a series of recommendations for further effective practical use of the modified bioinspired algorithm.

The General Conditions of the Algorithm: Considered modified bioinspired computing algorithm is a derivative model of "natural" algorithm of ant's colony activity.

It has been found that the marks of traces (pheromones) were applied to transmit information among ants about the route and share the gained knowledge about the direction, by moving ant leaves some marks (in different amounts) on the ground.

Isolated ant moves randomly. Ant, faced with the previously paved trace can find this route and follow it with high probability, thus, reinforcing the trace by own marks.

The collective behavior during search for a path is a form of autocatalytic behavior [3-7], in which the more ants follow to the known path the laid trace becomes more attractive for the next specimens.

Thus, the process is characterized by a positive feedback, where the probability of the ant's path increases with the growth of number of ants which previously chose the same path [3].

In our study, we used an imitation of the colony as a tool for optimization, in which the system is slightly different from the natural:

- Artificial ants have some memory;
- They are located in a space where time is discrete.

On Fig. 1a, the ants leave the point *A* which was considered as an anthill. The task for ants was to reach the food source located at the point *D* by the shortest route. If paved way contains the obstacle or there is a new path, an ant in point *F* should choose one of two variants either follows to a point *C* or to a point *E* (Fig. 1b). The selection is determined by amount of pheromone left by previous ants. A higher level marks on the path allows the ant choose the right direction.

To facilitate the conditions, lets both directions contain similar number of marks i.e. the choice of direction is equiprobable and the path *FED* is shorter the path *FCD*.

Initially, the ants reached point *F* may choose the path *FED* or *FCD* with equal probability. However, due to the path *FED* is shorter *FCD*, more ants will reach point *D* by this way per time unit and respectively and the path *FED* will contain more pheromone. Then the ants more likely will reach the food source and return to anthill using *FED* path (Fig. 1c). It should be noted that the ants will also use the path *FCD* but with less probability.

These arguments based on the fact that initial number of marks is similar. However, the conditions of the initial equality of pheromone traces are not required. For example, let's a new obstacle occurred on the path *FED* and extended longer the *FCD* path (Fig. 1d) then the ants passing *FCD* paths will faster reach the target point and a greater number of ants will pass this path increasing the pheromone trace per unit of time.

The result of all iterations is that the ants follow towards the anthill or food source will find more pheromone on the *FCD* path. It should be noted that once an ant have chose the path at the constant environment conditions they will choose this path again despite of variability of choice.

Mathematical Model of Algorithm: Let the number of cities is n, then the closed shortest path which only once cross all cities is the solution of the traveling salesman problem.

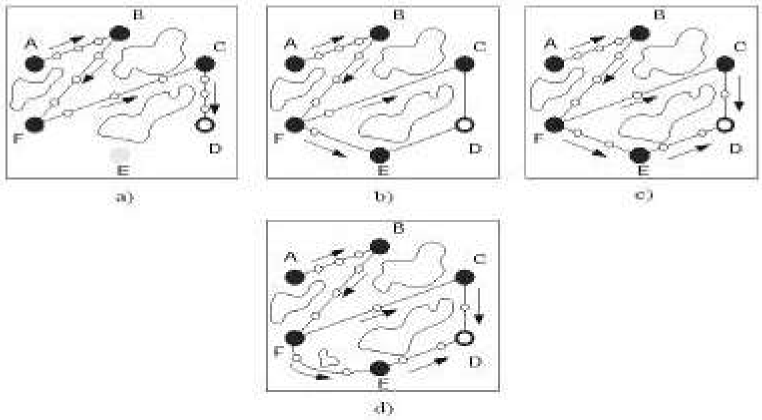

Fig. 1: Example of modeling the behavior of real ants

Distance from city i to city j – d_{ij} is determined by the following equation:

$$d_{ij} = (x_i + x_j)^2 + (y_i + y_j)^2 \cdot$$

The matrix with dimension (N,E) belongs to initial data, where N describes the order of cities and E is the distance between cities (the graph should be fully connected).

Let $b_i(t)$ ($i=1,... n$) is the number of ants in city i at time t while the total number of ants is m then:

$$m = \sum_{i=1}^{n} b_i(t) \cdot$$

Each Agent (Ant) Shall Satisfy the Following Conditions:

- He must choose a city with a probability depending on the distance and the number of traces in the specified direction;
- He must choose a city with a probability depending on the distance and the number of traces in the specified direction;
- He does not have the right to visit the traversed cities before the end of the route;
- If the route is passed the marks will activate.

Let $t_{ij}(t)$ is the intensity of marks in the interval (i,j) at time t then each ant chooses the next city at time t where it will be at time $t+1$.

Therefore, if we undertake m iterations of bioinspired algorithm for m ants in the interval (t, $t+1$), then each n-iteration of the algorithm (cycles) will result in the end of the route by an ant.

The intensity of the marks is calculated according to the formula:

$$\tau_{ij}(t+n) = \rho * \tau_{ij}(t) + \Delta\tau_{ij} \tag{2.1}$$

where ρ is a coefficient representing the increase of the intensity of the marks in the interval (t, $t+1$),

$$\Delta\tau_{ij} = \sum_{i=1}^{m} \Delta\tau_{ij}{}^{k} \tag{2.2}$$

where $\Delta\tau_{ij}^{k}$ is efficiency of interval (i,j) along whole route selected by k ant:

$$\Delta\tau_{ij}{}^{k} = \begin{cases} \dfrac{Q}{L_k}, \\ 0 \end{cases} \tag{2.3}$$

The first condition is satisfied if the k ant uses the direction (i,j) at time (t, $t+1$) overwise the second condition is satisfied.

Coefficient ρ must be less 1 to prevent unlimited accumulation of marks [3, 8-11]. In our experiment, we assumed that the intensity of traces $\tau_{ij}(t)$ at time $t=0$ is equal to some constant ρ.

To satisfy the constraints (each ant has to visit n different cities), each ant is associated with some data structure, called the *tabu*-list. It retains those cities which have been already visited at time t, i.e. the ant is prohibited visit the same city until the whole route will be passed.

When the route is passed, the values in the *tabu*-list are calculated for each of certain ant (i.e. the distance traversed by the ant is calculated). After *tabu*-list is nulled and the ant can move and choose further.

We determine $tabu_k$ as dynamically growing vector which contains the *tabu*-list of k ant and $tabu_k(s)$ is s element of the list (i.e. the s city which was visited by k ant in the certain route).

Let η_{ij} – the variable, inverse to distance ($\eta_{ij}=1/d_{ij}$), then the probability of transition of agent from city i to j – p_{ij} can be calculated by formula:

$$p_{ij}^{k}(t) = \begin{cases} \dfrac{\left[\tau_{ij}(t)\right]^{\alpha} * \left[\eta_{ij}\right]^{\beta}}{\sum\limits_{k \in allowed_k} \left[\tau_{ij}(t)\right]^{\alpha} * \left[\eta_{ij}\right]^{\beta}}, \\ \text{where } j \in allowed_k \text{ ? } allowed_k = \{N - tabu_k\} \\ 0 \end{cases} \tag{2.4}$$

where α, β are control parameters of the influence of visibility of marks.

Therefore, the possibility of transition is inversely dependent on the distance (i.e. closely located cities will be chosen with greater probability what reflects "greedy structural-heuristic approach") and intensity of marks at time t.

Description of Algorithm: The ants are randomly distributed between the various cities and the initial intensities of marks $\tau_{ij}(0)$ are equal to some constant ρ at time $t=0$. The first element of each ant's *tabu*-list is equal to the number of the city, where the cycle has started [12, 13]. Each ant moves from city i to city j by selecting the next city with a probability (2.4).

Intensity of marks $\tau_{ij}(t)$ provides information on the number of ants which have chosen the same route - the edge (i,j), in the past. The parameter η_{ij} uses heuristics of nearest city choice.

After the n-iterations, all ants will complete their route and their *tabu*-lists will be filled. The value of L_k is calculated for each k ant in this point and τ_{ij}^{k} calculated according to formula (2.3).

The shortest path found by ants ($\min\limits_{k} L_k$, $k = 1, \ldots, m$) will be saved. Then all the *tabu*-lists are nulled. This process is repeated until the counter of number of routes reaches the maximum determined by the user or all ants will pass this path.

Experimental Studies: The aim of experiments was search for an effective solution of the traveling salesman problem.

Full undirected final graph with *10* vertices was selected as the general test problem. The research program included the following steps:

- Investigate the relationship between the minimum value of the path and number of vertices in the graph (at all equal conditions);
- Determine the effect of varying quantitative value of the population size on finding of the minimum path in the graph (at all equal conditions);
- Identify the effect of a simultaneous change of the parameters α and β on the minimum path (at all equal conditions);
- Estimate the influence of change of the parameter Q on the minimum path;
- Determine the effect of increasing number of repeat cycles of the algorithm on the minimum path (at all equal conditions).

The study resulted in determination of empirical relationships, the variability limits of input parameters and elaboration of recommendations on their optimal choice.

- *Evaluation of dependence of the efficiency of the algorithm on the number of vertices in the graph*

In the experiment, let's consistently change the vertices in the graph from *10* to *29* with an interval *1*.

Initial data for the experiment were: $\alpha=5$, $\beta=1$, $Q=10$, the number of agents – *50* and the maximum number of cycles of the algorithm – *50*.

The dependence is shown on the graph (Fig. 2). X-axis corresponds to the number of cities and Y-axis to running time of algorithm in seconds.

Analysis of experimental data suggests that increase of the number of vertices in the graph from *10* to *29* results in an increase of the time spent on effective solution and takes values ??ranging from *5,8* to *130* seconds, respectively. The complexity of the algorithm is also increased.

- *Assessment of the influence of changes in population size on a minimal path in the graph*

During the experiment, the change in population size from *30* to *80* with interval *1* was studied.

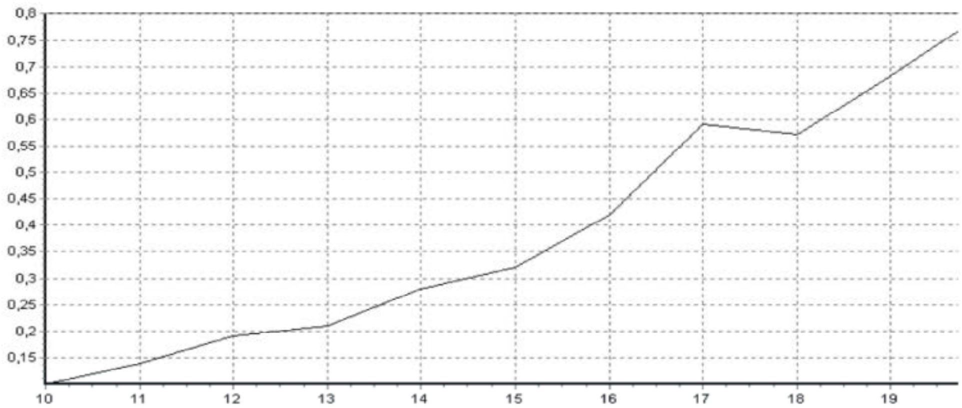

Fig. 2: Graph of dependence of time to solve the problem on the dimension of a graph

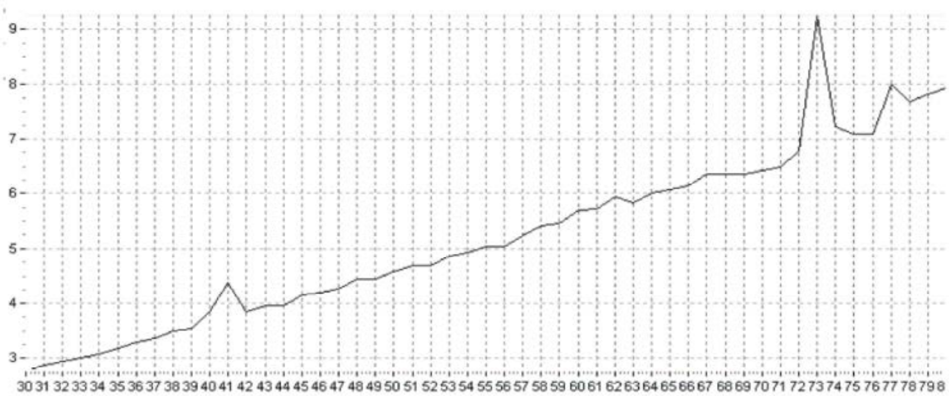

Fig. 3: Graph of dependence of time to solve the problem on the number of agents.

Initial data for the experiment were: α=5, β=1, Q=10 and the maximum number of cycles of the algorithm – *50*. The number of vertices in analyzed graph is fixed and equal to 10.

The dependence is depicted on the graph (Fig. 3). X-axis corresponds to the number of agents and Y-axis to running time of algorithm in seconds.

Analysis of experimental data suggests that gradual increase of population size results in an increase of the time spent on algorithm running.

Formula 2.2 shows that increase *m* (the number of specimens, agents in population) is followed by increase of τ_{ij} – efficiency of choice of the path from graph vertex *i* to *j*. Formula 2.3 shows that the value of L_k (minimum distance covered by *k* agent) should decrease (inverse relationship) at constant *Q* and increasing τ_{ij}. However, taking into account the probabilistic nature of the algorithm and the results of the modeling performed with the specified input parameters, the aptitude criterion can be represented as the graph on Fig. 4. X-axis corresponds to the number of agents and Y-axis to the length of the path.

Thus, an effective solution in this case is the number of agents equal to *36* and the path length will be the smallest value.

- *Assessment of the influence of alternating parameters α and β on the minimum path*

According to the formula (2.4), the gradual increase of the parameter α result in increase of $p_{ij}^k(t)$ (probability of transition of an agent from city *i* to *j* at time *t*). Similarly, the increase of $p_{ij}^k(t)$ occurs at selecting of the input parameter β as a variable parameter.

We considered the effect of increase of the parameter α (from *1* to *17* with interval *1*) on identification of the minimum cycle. Experimental studies were carried out at other equal conditions: studied graph contains *10* vertices, β=*1*, *Q*=*10*, the maximum number of algorithm cycles – *50* and the number of agents is *36*.

The obtained dependences are the functions of time and the aptitude criterion at varying parameter α will take the form as on Figs. 5 and 6. Fig. 5 shows that X-axis

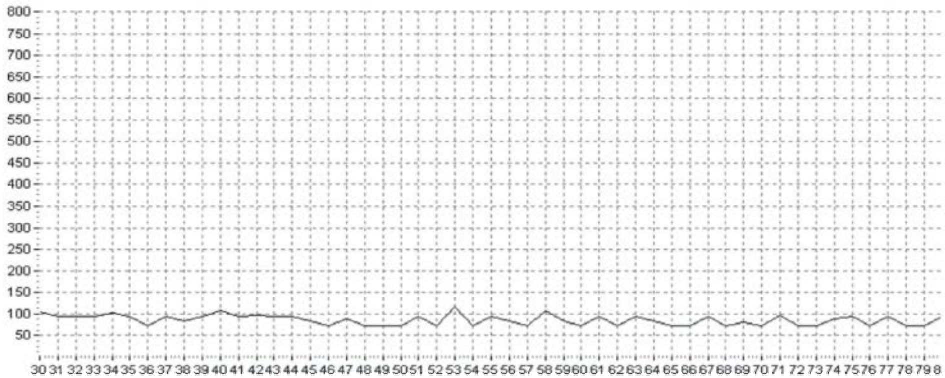

Fig. 4: Graph of dependence of path length on the change in the number of the agents

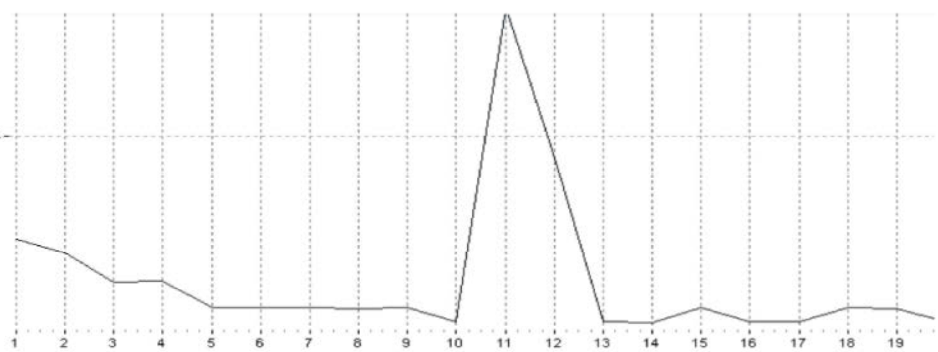

Fig. 5: Dependence of time for problem solving on α change.

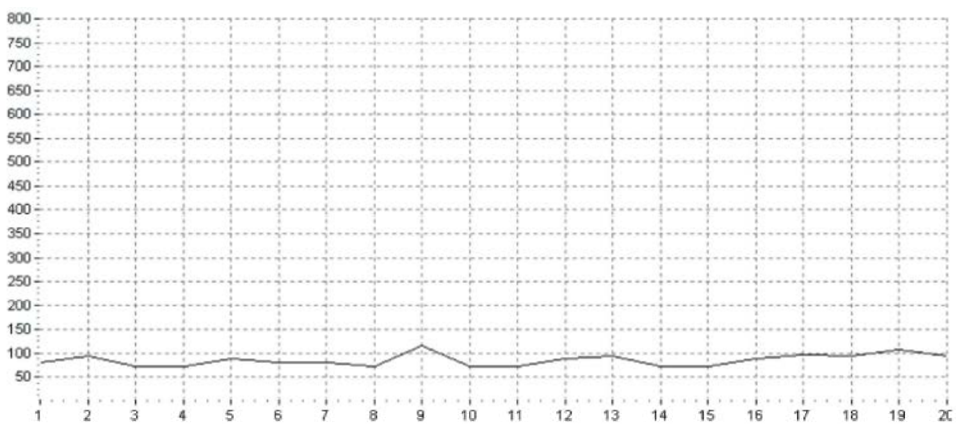

Fig. 6: Dependence of the path length depending on α change.

corresponds to α and Y-axis to running time of algorithm, in seconds. Fig. 6 indicates that X-axis corresponds to α and Y-axis to the length of the path.

The results (Figs. 5 and 6) demonstrate that the effective solution of this problem is achieved by increasing the values of α (in interval from *15* to *17*) and is *124*.

We considered the effect of increased parameter β (from *1* to *10* with interval *1*) on the minimum cycle identification. Experimental studies were carried out at other equal conditions: studied graph contains *10* vertices, $\alpha=15$, $Q=10$, the maximum number of algorithm cycles – *50* and the number of agents was *36*.

The obtained dependences are the functions of time and the aptitude criterion at varying parameter β will take the form as on Figs. 7 and 8. Fig. 7 shows that X-axis corresponds to β and Y-axis to running time of algorithm, in seconds. Fig. 8 indicates that X-axis corresponds to β and Y-axis to the length of the path.

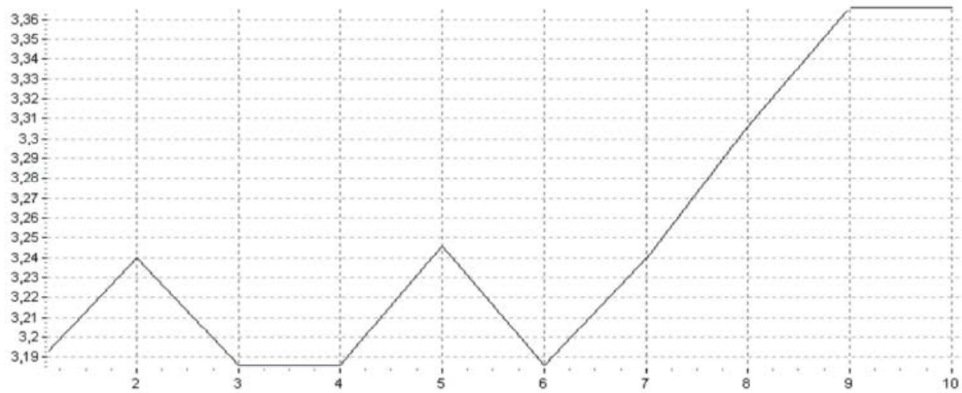

Fig. 7: Dependence of time for problem solving on β change

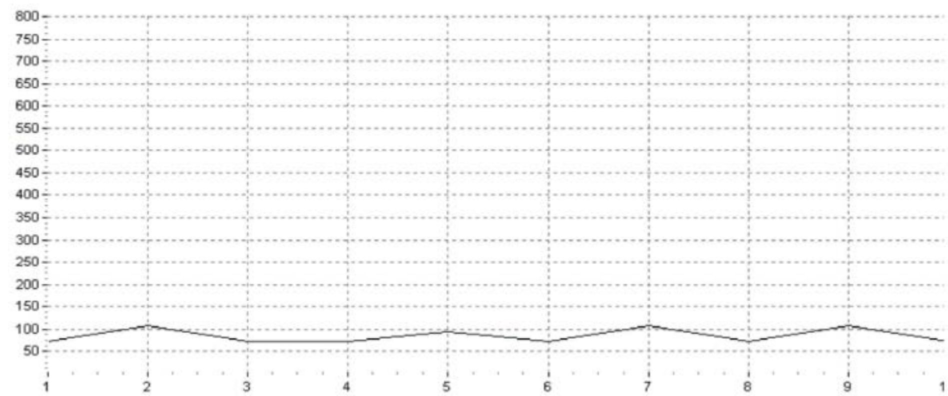

Fig. 8: Dependence of the path length on β change.

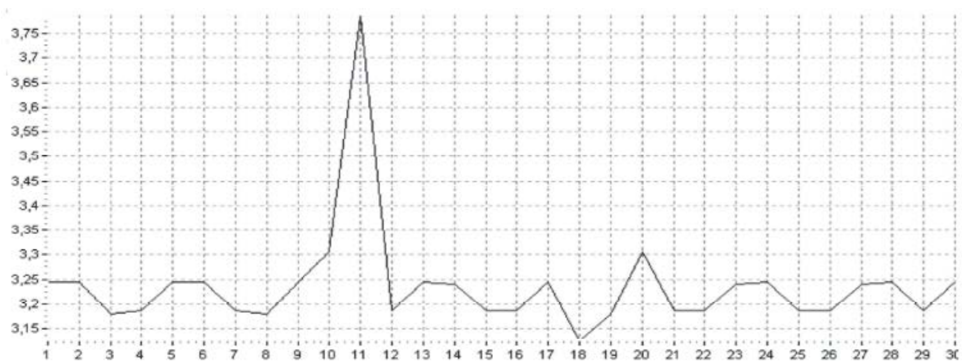

Fig. 9: Dependence of time for problem solving on Q change.

The results (Figs. 7 and 8) show that the sufficiently "good" solution to the problem can be achieved by alteration of β from *1* to *4* and was *124*.

- *Assessment of the influence of alternating parameters Q on the minimum path*

According to the formula (2.3), the gradual increase of Q and fixed L_k results in increase of τ_{ij}^k – efficiency of path choice from vertices *I* to *j*. The growth of τ_{ij}^k results in increase of the probability of $p_{ij}^k(t)$ according to the formula (2.4).

Thus, the increase of the input parameter Q results in accumulation of a larger number of marks along the most effective routes in the graph.

Experimental studies were carried out when Q varied from *1* to *30* and the other equal conditions: studied graph contained 10 vertices, $\alpha=15$, $\beta=3$, the maximum number of algorithm cycles – *50* and the number of agents was *36*.

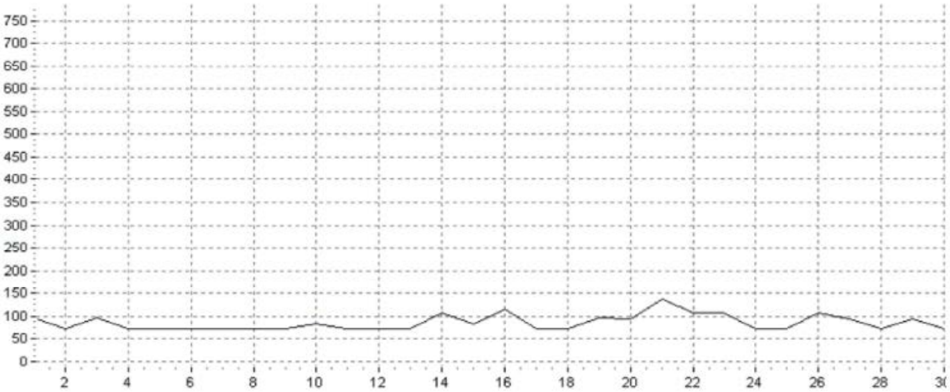

Fig. 10: Dependence of the path length depending on Q change.

Fig. 11: Dependence of time for problem solving on the number of iterations of the algorithm

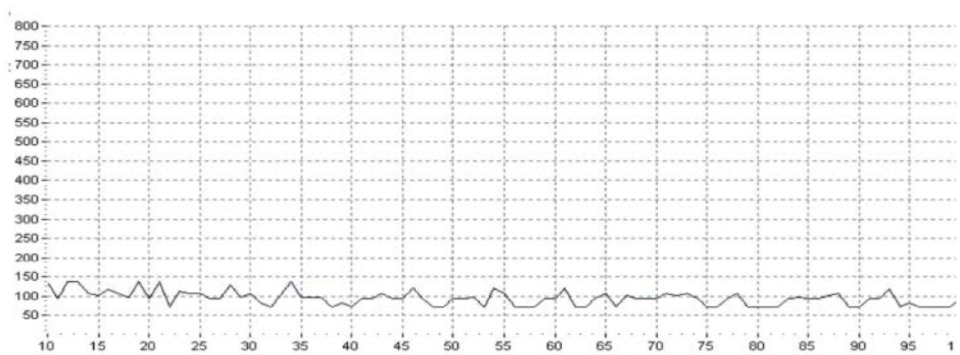

Fig. 12: Dependence of the path length on the number of iterations of the algorithm

The obtained dependences are the functions of time and the aptitude criterion at varied parameter β and depicted on Figs. 9 and 10. On Fig. 9, X-axis corresponds to Q and Y-axis to running time of algorithm, in seconds. Fig. 10 indicates that X-axis corresponds to Q and Y-axis to the length of the path.

Figs. 9 and 10 show that the alteration of parameter Q has no significant influence on both functions of time and aptitude criteria. To avoid the stagnation state, the value of Q should be chosen within an interval from *21* to *30*.

- *Assessment of the influence of alternating number of algorithm cycles on the problem solution*

The experiment included the following conditions: possible number of cycles varied from *50* to *200* with an interval *1*, the graph contained *10* vertices, $\alpha=15$, $\beta=3$, $Q=30$ and the number of agents was *42*. An effective interval of solutions at the number of iterations from *125* to *135* was the result of study.

The obtained dependences are the functions of time and the aptitude criterion at varied parameter β and depicted on Figs. 11 and 12. On Fig. 11, X-axis corresponds to the number of iterations of the algorithm and Y-axis to running time of algorithm, in seconds. Fig. 12 indicates that X-axis corresponds to the number of iterations of the algorithm and Y-axis to the length of the path.

Identification of the current record occurs in every iteration of the graph and followed by a new iteration of the algorithm. The new "satisfactory" result was compared with the previous and best of these was remembered. Thus, the gradual increase of the number of iterations resulted in the experience accumulation i.e. the agents have a greater chances to choose the most effective direction.

CONCLUSIONS

This article represents the experimental study of development of modified bioinspired algorithm for solving of the traveling salesman problem. The simulation results suggest significant advantage of the considered algorithm compared with existing analogues. Fast identification of the effective solution regardless of the range of variable parameters was found as the main feature of the modified bioinspired algorithm.

This indicates the flexibility and stability of the studied method. Numerical experiment determined the empirical dependences, the intervals of the input parameters and allowed the elaboration of several recommendations on the optimal choice. The series of tests and experiments have refined the theoretical estimates of the time complexity of bioinspired algorithm: at the best case, the time complexity of algorithms is ≈ $O(nlogn)$ and in the worst case – $O(n^3)$.

REFERENCES

1. Abraham, A., G. Grosan and V. Ramos, 2006. Swarm Intelligence in Data Mining. Berlin, Heidelberg: Springer Verlag, pp: 267.
2. Kureichik, V.M. and V.V. Kureichik, 2001. Genetic Search-Based Control. Automation and Remote Control, 62(10): 1698-1710.
3. Dorigo, M., V. Maniezzo and A. Colorni, 1996. Ant System: Optimization by a Colony of Cooperating Agents. In IEEE Transactions on Systems, Man and Cybernetics-Part B, 26(1): 29-41.
4. Kureichik, V.M., S.P. Malioukov, V.V. Kureichik and A.S. Malioukov, 2009. Algorithms for Applied CAD Problems. Berlin, Heidelberg: Springer-Verlag. 212: 236.
5. Qing, He, Xiu-Rong Zhao, Ping Luo and Zhong-Zhi Shi, 2007. Combination methodologies of multi-agent hyper surface classifiers: design and implementation issues. In Second International Workshop, AIS-ADM 2007, Proceedings. Berlin Heidelberg: Springer-Verlag, pp: 100-113.
6. Kureichik, V.M. and V.V. Kureichik, 2006. A Genetic Algorithm for Finding a Salesman's Route. Journal of Computer and Systems Sciences International, 45(1): 51-58.
7. Shi, Z.Z.H., Y. Zhang, Y. Cheng, Q. Jiang Sheng and Z. Zhao, 2004. MAGE: an agent-oriented programming environment. In Proceedings of the IEEE International Conference on Cognitive Informatics. New York, pp: 250-257.
8. Kureichik, V.V., V.M. Kureichik, L.A. Gladkov and P.V. Sorokoletov, 2009. Bioinspired Optimisation Algorithms. Moscow: Fizmatlit, pp: 384.
9. Kureichik, V.M., V.V. Kureichik and L.A. Gladkov, 2010. Genetic Algorithms. Moscow: Fizmatlit, pp: 368.
10. Kureichik, V.M., V.V. Kureichik and S.I. Rodzin, 2012. Evolutionary Calculations: Theory, Models, Algorithms and Parallelism. Saarbrucken, Germany: LAP LAMBERT Academic Publ. pp: 330.
11. Kureichik, V.V., V.M. Kureichik and S.I. Rodzin, 2009. The Concept of Evolutionary Calculations Inspired by Natural Systems. In Bulletin of Southern Federal Univercity: Technical Sciences. Taganrog: TTI YuFU, 4(93): 16-25.
12. Kureichik, V.M., V.V. Kureichik and P.V. Sorokoletov, 2007. Analysis and a Survey of Evolutionary Models. Journal of Computer and Systems Sciences International, 46(5): 779-791.
13. Kacprzyk, J., V.M. Kureichik, S.P. Malioukov, V.V. Kureichik and A.S. Malioukov, 2009. Evolutionary models of decision making. Studies in Computational Intelligence, 212: 23-114.