УДК 519.178

Алгоритм разрешения конфликтов в задаче выбора оптимальных маршрутов группы агентов на плоскости

Кирсанов Михаил Николаевич
НИУ «МЭИ»
Профессор

Аннотация
Для решения задачи прокладки кратчайших путей по ребрам взвешенного графа с учетом иерархии агентов по приоритету достижения вершин применяется муравьиный алгоритм. В параметр, определяющий выбор очередной вершины, вводится дополнительный коэффициент, уменьшающийся после возникновения конфликта между агентами, прибывшими в эту вершину одновременно. Алгоритм строится в системе компьютерной математики Maple и может найти применение для управления группы мобильных роботов.
Ключевые слова: граф, маршрут, робот, конфликт, агент, Maple, управление роботами

Algorithm for resolving conflicts in the task of selecting the optimal routes for a group of agents in the plane

Kirsanov Mikhail
NRU «MPEI»
Professor

Abstract
To solve the problem of laying the shortest paths along the edges of a weighted graph, taking into account the hierarchy of agents, the ant algorithm is used to achieve the vertices. In the parameter that determines the choice of the next vertex, an additional coefficient is introduced, decreasing after the conflict between agents arrived at this vertex simultaneously. The algorithm is built in the system of computer mathematics Maple and can find application for the management of a group of mobile robots.
Keywords: graph, route, robot, conflict, agent, Maple, management of a robots.

Formulation of the problem. The task is to find the optimal interaction of an organized group of objects (robots, «agents») when they perform a certain planned work. In particular, the specific task for a group of objects in the case under investigation is to move them along a certain plane with the marked points forming the vertices of some incomplete graph (Fig. 1). The start and end points of each object are different. In the example, we consider the case of *two agents* with initial points in A_0, B_0 and finite in A_k, B_m. The shortest paths from the initial points to the finite ones are found, taking into account the limitation of locating the agents at the vertices. This means that simultaneously (with a given time error $\tau_j, j = 1,...,M$) in one vertex (or «city», by the tradition of the *traveling salesman problem*), two or more agents can not be located. The agent group has a hierarchy of significance. First of all, an agent with greater significance is admitted into the city. In addition, in cities-peaks there are also differences in the need for their passage. To solve the problem, the ant algorithm was chosen by Marco Dorigo [1].

The idea [2] of optimization of transport routes on the basis of the introduction of feedbacks of various types (positive and negative) in the form of some algorithm (figuratively called *ant*) for the first time in the Russian-language literature was described by S. Shtovba [3, 4]. In the future, the effectiveness of this approach to optimization problems has been repeatedly proven. In [5], some modification of the algorithm of the ant algorithm is presented with reference to the problems of optimization of transport routes in a network of large city streets with two-way traffic. The classic *traveling salesman problem* was solved. Three metric versions are used: with time, distance and speed estimation. In solving the problem, both synchronous and asynchronous movement of ants –agents were allowed. In [6], based on the ant algorithm with the use of dynamic variables (reducing the space for finding solutions), a general scheme for modeling optimization and optimal planning of irrigation and fertilization of annual crops in agricultural practice was developed. The problem of adding a third parameter to the ant algorithm used to solve optimization problems is discussed in [7]. In a series of papers [8-12] various variants of the ant algorithm, its applications and analysis are considered. The solution of the classic traveling salesman problem with the help of the ant algorithm is implemented in the program [13] in Maple.

Based on this program in this paper, the problem of resolving the conflicts of tracing a multi-agent system on a plane is solved. The estimation of the priority of the vertex for continuation of the route is determined by the value

$$P_{ij,k} = \frac{\tau_{ij}^{\alpha} \lambda_{ij}^{\beta} s_j^{\zeta}}{\sum_m \tau_{im}^{\alpha} \lambda_{im}^{\beta} s_m^{\zeta}} \ , \tag{1}$$

where τ_{ij} — the amount of pheromone left by ants on the arc [i, j], λ_{ij} — is the reciprocal of the weight (length or conditional cost) of the arc. In comparison with the known algorithm [1-4], here we give one more parameter s_j, corresponding to the conditional availability of the vertex *j*. Accessibility is regulated by the interaction of agents in the process of movement. If at some time in the vertex *j* there are two agents, then for an agent of lesser significance, this parameter, after passing the entire path, becomes an agent of greater significance, decreases. The coefficients α, β, ζ determine the priority of the corresponding quality. If, for example $\alpha = 0$, the agents are guided only by the length of the path when choosing a possible path. If $\beta = 0$ the most valuable information is the experience of previous ants (the level of pheromone). To resolve the conflict (the order of approach to the vertex) the parameter ζ must be different from zero. The value of this parameter determines only the choice of a spare city (top) for an agent of lesser importance, for which entry is already prohibited in this city.

Algorithm. In fact, the proposed algorithm is very close to the algorithm for solving the *Hamilton problem*. However, while the complete graph was considered in the Hamilton problem, in this formulation the problem is complicated by the fact that the graph is a *network* with a *source* and *drain point* for each agent. The start and end points can be different, and in some cases the routes are not implemented because of possible the deadlocks (hanging) vertices. For simplicity, in this solution, hanging vertices will not be considered. The main idea of the algorithm for a group of agents with the priority of visiting vertices is to consistently traverse paths, starting with the agent with the highest rank. In the process of passing (still not passed by anyone the graph) of vertices the agent observes the visiting time of each vertex. Time depends on the speed of the agent and the weight of the edges (the distance between the vertices). After that, some of the vertices have the status of marked, other part — no (Figure 2). The next agent, passing through the graph (according to the usual rule of selecting the next vertex of the ant algorithm with a mark of the pheromone passed by the edges, is inversely proportional to the time of passage of the entire path), also notes the time of passage of the vertices.

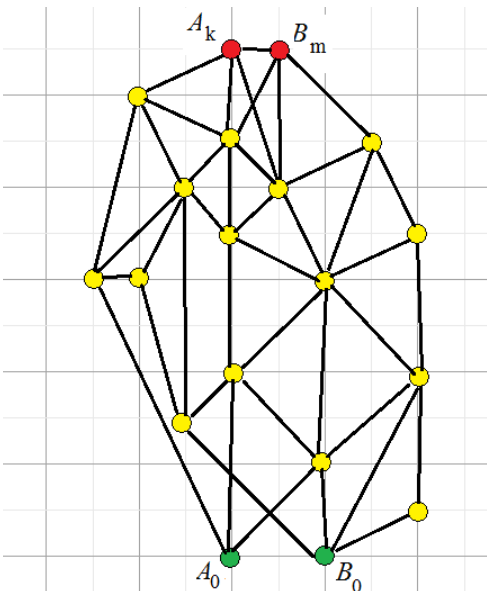

Figure 1 — Source graph

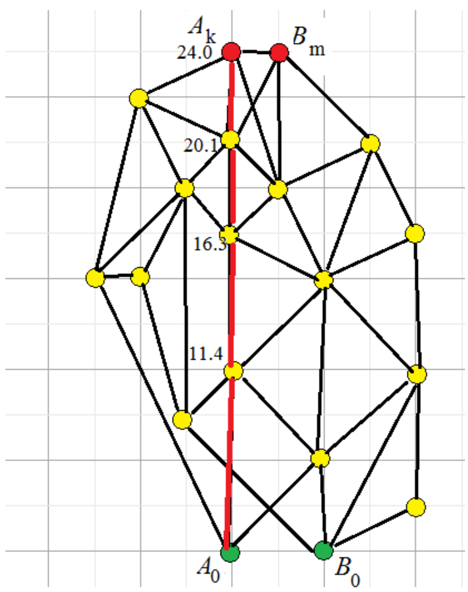

Figure 2 — Step 1. Agent *A* route

If the time of some vertex is identical to the accepted error (Fig. 3), then the status of this vertex decreases, and the route of this agent is re-laid (Fig. 4) to the accuracy of the accepted error. It should be noted, however, that the term «route» is understood here taking into account the many iterations, similar to the classical problem of the ant algorithm, where due to the randomness of selecting the next vertex, regulated by the value (1), an optimal route is constructed. The number of iterations depends on the number of routes, the presence of *«elite» ants*, the parameter of «*drying out*» of pheromone [3, 13].

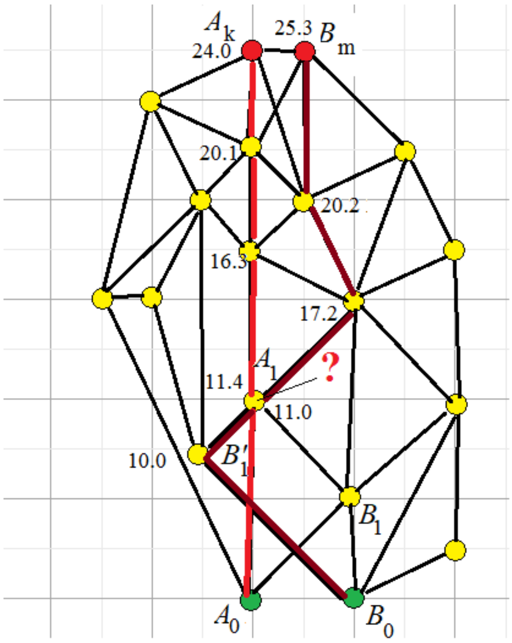

Figure 3 — Agent *B* route

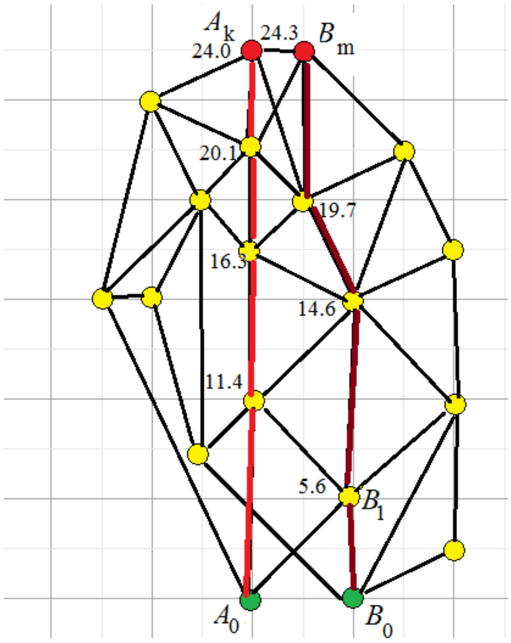

Figure 4 — Conflict resolution

Figure 3 shows several «unnatural» direction of agent *B*. Instead of heading to the top B_1, the agent turned to the top B'_1. However, this solution was obtained as a result of an integral evaluation of the entire path based on the results of many iterations of the ant algorithm. «Conflict» at the top A_1 with Agent A, led to the fact that by reducing the accessibility of this vertex, Agent B «corrected» its route by selecting this time the previously rejected route to the top B_1. We note that the proposed algorithm is designed for knowledge of the coordinates and characteristics of all cities by agents (a complete «map» is known). In the case of local choice of directions, without preliminary information about all vertices, differential methods should be used, for example, based on the *Bellman principle* with an estimate of the additive or multiplicative objective function with the possibility of introducing corrections to the agent conflict at the vertices of the graph.

The work was supported by the RFBR grant No. 16-01-00429 a.

References

1. Dorigo M. Optimization, Learning and Natural Algorithms, PhD thesis, Politecnico di Milano, Italia, 1992.
2. Dorigo M., Stützle T. Ant colony optimization // A Bradford book, 2004.
3. Штовба С.Д. Муравьиные алгоритмы // Exponenta Pro. Математика в приложениях. 2003. №.4. С. 70-75.

4. Shtovba S.D. Ant algorithms: theory and applications // Programming and Computer Software. 2005. Т.31. №4. С.167-178.
5. Danchuk V., Bakulich O., Svatko V. An Improvement in ant Algorithm Method for Optimizing a Transport Route with Regard to Traffic Flow // Procedia Engineering. 2017. Т.187. С.425 – 434
6. Nguyen D.C., Ascough J.C., Maier H.R., Dandy G.C., Andales A.A. Optimization of irrigation scheduling using ant colony algorithms and an advanced cropping system model // Environmental Modelling and Software, 2017. Т. 97, №11. С. 32-45.
7. Kirsanov M.N., Shpitonkova E.A. Simulation of ant algorithm using maple with adding additional parameter // Системы компьютерной математики и их приложения. Материалы XIV Международной научной конференции, посвященной 90-летию профессора М. Б. Балка. Выпуск 14. Смоленск. Издательство СмолГУ 2013. С. 57-59.
8. Лебедев Б.К., Лебедев О.Б. Моделирование адаптивного поведения муравьиной колонии при поиске решений, интерпретируемых деревьями // Известия ЮФУ. Технические науки. 2012. № 7 (132). С. 27-34.
9. Курейчик В.М., Лебедев Б.К., Лебедев О.Б. Гибридный алгоритм разбиения на основе природных механизмов принятия решений // Искусственный интеллект и принятие решений. 2012. С. 3-15.
10. Лебедев В.Б. Метод пчелиной колонии в комбинаторных задач на графах // Тринадцатая национальная конференция по искусственному интеллекту с международным участием КИИ-212. Труды конференции. М.: Физматлит, 2012. Т. 2. С. 4l4–422.
11. Лебедев О.Б. Трассировка в канале методом муравьиной колонии // Известия ЮФУ. Технические науки. 2009. № 4 (93). С. 46–52.
12. Лебедев О.Б. Построение дерева Штейнера на основе метода муравьиной колонии // Труды конгресса по интеллектуальным системам и информационным технологиям «AlS-IT’09». Научное издание в 4-х т. Т. 1. М.: Физматлит, 2009. С. 58-65
13. Кирсанов М. Н. Задачи по теоретической механике с решениями в Maple 11. М.: Физматлит, 2010. 264 с.