ATLANTIS PRESS
Advances in Intelligent Systems Research, volume 174
Proceedings of the 8th Scientific Conference on Information Technologies for Intelligent Decision Making Support (ITIDS 2020)

The Ant Colony Method to Support Decision-Making on the Appointment of Employees

Sudakov V.A. *
NL Applied Modeling
Plekhanov Russian University of Economics
Moscow, Russia
sudakov@ws-dss.com

Titov Yu.P.
Institute No. 3 Control Systems
Informatics and Power Engineering
Moscow Aviation Institute
(National Research University)
Moscow, Russia
kalengul@mail.ru

***Abstract*—Modern planning tools for the development and production process operate with schedules. The main disadvantage of scheduling is the difficulty in determining when to complete a task. The paper considers an approach that makes it possible to estimate the execution time of a task in the form of a fuzzy function, using the estimated representation of employees as input data. The procedure for calculating the time of a task is described, provided that several employees are assigned to it and the time required for their interaction is taken into account. The arising problem of the rational appointment of employees is proposed to be solved using a modification of the ant colony method**

Keywords—Fuzzy sets, assignment problem, ant colony method, work time

I. Introduction

The paper deals with the task of assessing the time of execution of individual tasks and work in general. Currently, the project manager himself determines the assessment of the execution time of the entire project and its individual stages based on his experience and understanding of the complexity of the work. [1,2] This approach is well suited for analogy projects. For innovative projects, determining the time to complete the work is difficult and usually relies on some expert assessments of the employees themselves. For example, agile methodology is used for software development [3-5].

When using employee assessments, there is a problem of inaccuracy in determining the time for completing a task by one specific employee. To solve this problem, it is proposed to use fuzzy sets. As a result of the survey, it is possible to construct a fuzzy set of task execution times for each employee. In this case, for each employee, several fuzzy sets can be assigned for each task that he can perform. [6-12] As a result, the problem arises of the rational assignment of employees to tasks in order to minimize the total time of the project.

For the case when only one employee can be assigned to one task, the assignment problem is solved by dynamic programming methods, since it has a separable function and additive constraints. For optimization on a fuzzy set, a defuzzification procedure is used to obtain a specific value of the optimization criterion.

If several employees can be assigned to one task, then the calculation of the task execution time cannot be performed by ordinary mathematical methods. [13] It is necessary to apply the procedure for generalizing fuzzy sets defined on different bearing sets, and take into account the time spent on interaction of employees. As a result, the solution to the problem of assigning employees is impossible using dynamic programming methods. The paper proposes the use of a modification of the ant colonies method to solve the assignment problem.

II. Generalization of Fuzzy Sets

The main problem in the generalization of fuzzy sets is in the various bearing sets and the operation of determining the resulting bearing set. So, to generalize the fuzzy sets describing the task execution time, the carrier set is not considered the task execution time, but the employee's productivity. To do this, calculate the reciprocal of the execution time and the corresponding value of the membership function for a fuzzy set.

$$\mu_{i,j,k} = 1 - \nu_{i,j,k}; \quad (1)$$

$$p_{i,j,k} = 1/t_{i,j,k}; \quad (2)$$

For two fuzzy employee productivity functions, the generalization operation is to find the value of the membership function for each value of the generalized employee productivity, calculated as the sum of the productivity of all employees assigned to this task. But due to the sum of two or more continuous numbers, the calculation of the value of the membership function that satisfies the generalization condition is almost impossible.

$$\mu_j(p_j) = \max_{p_j=\sum_{i \ni I_j} p_{i,j}} \left(\min_i \left(\mu_{i,j}(p_{i,j})\right)\right) \quad (3)$$

To find the value of the membership function, it is proposed to consider the problem in a different form. It is proposed to sort out not the values of productivity (bearing set), but the values of the membership function and for each such value to determine the productivity of workers. [14,15] Such a statement is possible if the conditions of continuity and monotony of the membership functions of fuzzy productivity functions are met, which in most cases is performed. For formula (3), the maximum value is taken with the same values of the accessory functions. As a result, the task of calculating a fuzzy function of a generalized membership function can be reduced to finding performance values when an employee performs a task for a certain value of the membership function. As a result, the procedure for generalizing membership functions can be represented as:

$$\mu_{i1,j}(p_{i,j}) = \mu_{i2,j}(p_{i,j}) = \max_{p_j=\sum_{i\ni I_j} p_{i,j}} (\min_i(\mu_{i,j}(p_{i,j}))) = \mu_j(p_j): \forall i1, i2; \quad (4)$$

$$p_j = \sum_{i\ni I_j} p_{i,j}: \mu_j(p_j) = \mu_{i,j}(p_{i,j}), \text{для } \forall i \ni I_j, \forall \mu_j(p_j) \in (0..1), \quad (5)$$

After obtaining the generalized membership function, the fuzzy function of the task execution time by a group of employees is calculated.

III. Calculation of the Time to Complete the Work and Accounting for Employee Interaction

After calculating the fuzzy function of the execution time of each task, you can calculate the execution time of the entire job or stage. First, you need to consider the interaction of workers assigned to one task. If we imagine the most suboptimal case of interaction, then it can be described by the phrase: "If you want to do well, do it yourself." In this case, a more experienced worker, i.e. an employee who does work faster teaches an inexperienced person to his conditional level.

$$t_{вз} = \max_t(\nu_{i1,j}(t_{i1,j}) - \nu_{i2,j}(t_{i2,j})), \text{для } \forall\, i1, i2 \ni I_j \quad (6)$$

If we consider all possible interactions between employees, then this time will greatly affect the total time to complete the task. Interaction between workers can be represented as a fully connected graph of interaction G (V, E), where V is the set of node corresponding to workers, and E is the set of arcs corresponding to interactions between them and determining the required time. In this graph, you can consider various paths, for example, the Hamiltonian path that determines the most optimal interaction along the chain. But the best solution assumes some organization of the process of working on the task and application, for example, a leader or team leader. Then it is worth considering arcs from the manager to other employees. It should be taken into account that the time for interaction between employees should be taken with a certain weighting coefficient k_вз from the interval (0; 1). But it is possible to specify the interaction of workers in the form of a fuzzy function.

Further calculation of the work execution time is determined by the ability to manage the sequence of tasks. If there is a sequence of tasks: a schedule or a Gantt chart, then it is possible to use fuzzy task time functions to calculate the fuzzy time to complete work. At the same time, the interaction time of employees shifts the task execution time upward. A simpler approach is to defuzzify a fuzzy function and obtain a specific value for the task execution time. Among the many defuzzification algorithms, it is recommended to use different a-levels. If you do not have a specific task sequence, for example with SCRUM Sprint, you can determine the maximum time to complete tasks or optimize assignments until certain limits on the Sprint time are reached.

As a result, the entire algorithm for working with fuzzy sets is shown in Figure 1. It should be noted that this algorithm can only work if there is information about the assignment of employees to tasks, i.e. either jointly with the manager deciding this issue or using optimization tools. Moreover, most often these tools are decision support systems that take into account the preferences of an expert and search for not only optimal, but also rational decisions. This article discusses the possibility of applying a modification of the ant colony method to find rational solutions in the problem of assigning employees.

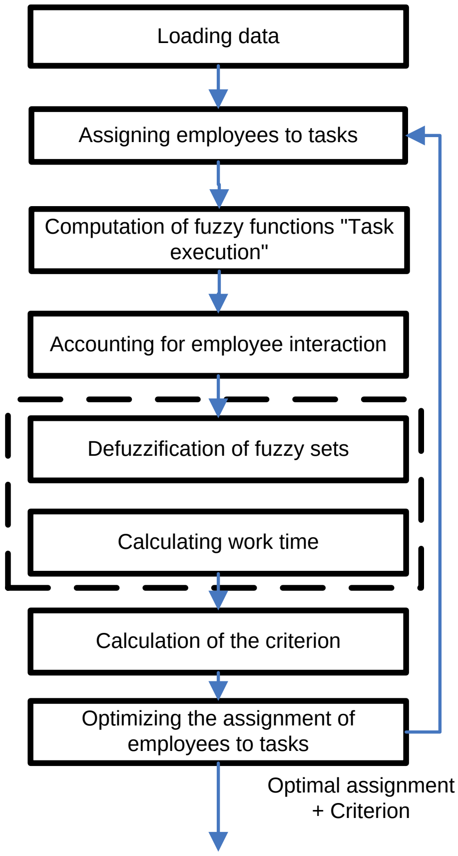

Fig 1. The process of calculating the time of work by a group of employees using fuzzy sets.

IV. Modification of the Ant Colony Method for Solving the Assignment Problem

Based on the results of the algorithms proposed above, it is possible to calculate the execution time for a specific assignment of employees to individual tasks. To determine the rational assignment of employees to tasks, it is proposed to apply a modification of the ant colony method. The original ant colony method was developed for finding the traveling salesman path in a graph. [16] But to solve the assignment problem it is necessary to use a special decision graph (Fig. 2). [17-19] For each employee and the task that he can perform, a node is created. All nodes belonging to one employee are combined into a layer. Arcs connect node from adjacent layers. In this case, the order of the layers is not important; since the end of the search for a solution is the path from the top of the first layer to the last layer. It should be noted that in the decision graph for each employee it is necessary to add a node that determines the absence of an employee's assignment to the task. This is largely necessary due to the presence of accounting for the time of interaction of employees, i.e. adding a new employee can lead to a delay in the process of solving the problem. Agents (Ants) begin their movement from an imaginary node connected by arcs to all node of the first layer, i.e. first employee.

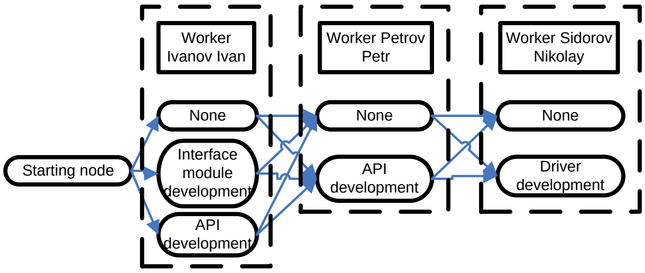

Fig 2. The structure of the decision graph.

After passing the completion of the movement of the agent, his path describes the assignment of each employee to a specific task. For the assigned assignment, the time to complete the work is determined. The probabilistic choice between the paths for moving an agent from one node to an adjacent node is determined based on the (Pheromone) weights of neighboring node. The choice of the node of the graph as the weights carrier is primarily due to the need to determine the specific assignments of employees, and not the relationships between assignments. In addition, this approach allows us to reduce the running time of the algorithm, since the number of node in the decision graph is much less than the number of arcs. The procedure for evaporating weights from the node of the graph is similar to the original procedure. For the algorithm to work, not one agent, but a certain group passes through the decision graph in one iteration. After all the agents from the group pass through the decision graph and calculate the execution time for the found paths, the weights of the node of the decision graph are changed. Depending on the work execution time, an additional weight is added to each node from the path traversed by the agent, calculated by the formula: The shorter the work execution time, the more weights will be added to the corresponding node and the more likely the agents will choose these nodes. This procedure allows you to iteratively converge to an optimal solution. The sequence of stages of the algorithm is shown in Figure 3. [20,21]

The proposed algorithm makes it possible to find rational assignments of employees to tasks. To speed up the search for rational solutions, various methods can be used: setting the optimal parameters of the ant colony algorithm, using modifications of the ant colony method, for example, a ranked modification, and others. In addition, it is worth noting that for the ant colony algorithm the procedure for stopping its operation is blurred, for example, it can stop at a certain number of iterations or when finding a solution that satisfies the constraints. In the general case, the operation of the algorithm does not need to be stopped.

It should also be noted about the problem of "looping" the method of ant colonies, i.e. states when the distribution of weights on the decision graph does not allow agents to move along new paths. As a result, the algorithm cannot find a solution different from the available one. To solve this problem, it is proposed to reset the decision graph when certain criteria for the operation of the algorithm are achieved. As a result, the operation of the algorithm can occur without a specific stopping moment

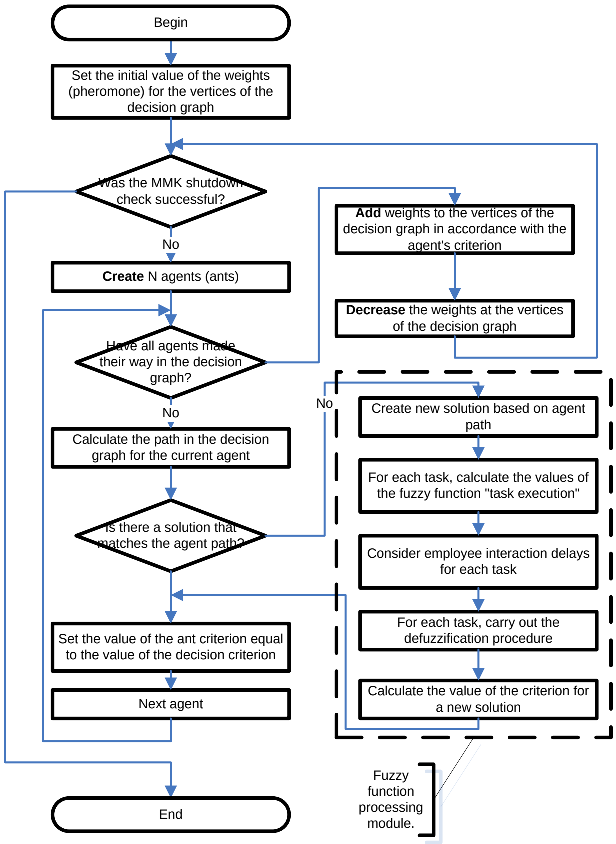

Fig 3. Algorithm for the modification of the ant colony method

V. Realization and Testing

To test the proposed algorithm, the problem of assigning 35 employees to 15 tasks was considered. Each employee could perform a different number of tasks. In total, the test examined more than 180 accessory functions "task completion by a specific employee". To solve this problem, software has been written that allows not only to search for a solution that is rational in terms of the execution time, but also to enumerate the parameters of the ant colony method to optimize the search speed for a solution. As a result, an optimal solution to the assignment problem was found in the form of assigning employees to tasks, as well as estimating the time to complete the entire work.

During testing, the following patterns were revealed in the setting by the parameter of the ant colonies method: The number of agents in one group should be selected from the interval (Number of decision graph layers; Number of decision graph layers * 2). It is recommended to take the evaporation coefficient in the range (0.8; 0.95). (Fig. 4) In this case, it is recommended to use a ranged algorithm for entering weights with a parameter equal to the number of agents / 4. (Fig. 5) On graph 4 and 5, the dashed line marks the confidence interval of the estimate of the mathematical expectation, calculated from the results of 500 runs

To solve the looping problem, the following states of the method were considered:

- No new solutions were found during the iteration. This method only works if multiple solutions are stored. It is easy to perform the check algorithmically; it is enough to compare the dimension of the set of solutions before and after the iteration. This approach resets the decision graph more often than others, since it does not take into account that agents can choose different paths during iteration, but only takes into account the process of increasing the set of considered solutions. (dark gray squares in the graphs 6)
- The mathematical expectation of the values of the decision criteria found at the iteration is equal to the value at the previous iteration. This criterion makes it possible to judge that the agents choose the same paths between iterations. With a high probability, it can be assumed that further, at the next iteration, the agents will choose the same paths. (light gray triangles in the graphs 6)

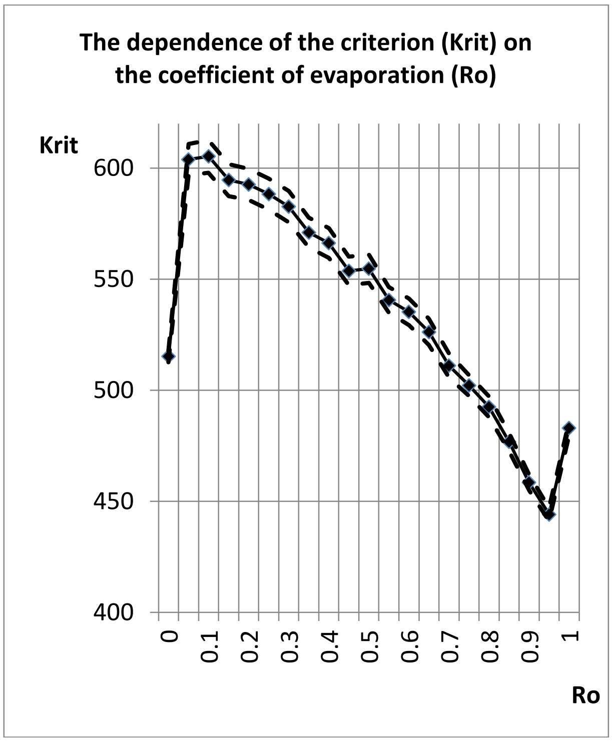

Fig 4. Determination of the optimal parameters coefficient of evaporation for the modification of the ant colony method.

The proposed algorithms fight against the looping problem for non-optimal parameters of the ant colony method. For the parameter responsible for the number of agents in one group, this influence is not so significant, with the number of agents from 20 to 40, the percentage of successful runs increased by only 10%, and at the same time, most of the runs ended in "looping". This is due to the resulting low number of possible considered solutions, for example, for 20 agents with the limitation on looping 1000 iterations, the maximum number of considered solutions is 20,000. There is a limit to the number of necessary solutions for a confident search for the optimal one. This problem can be solved not only by increasing the number of agents, but also by increasing the number of iterations for "looping" the algorithm

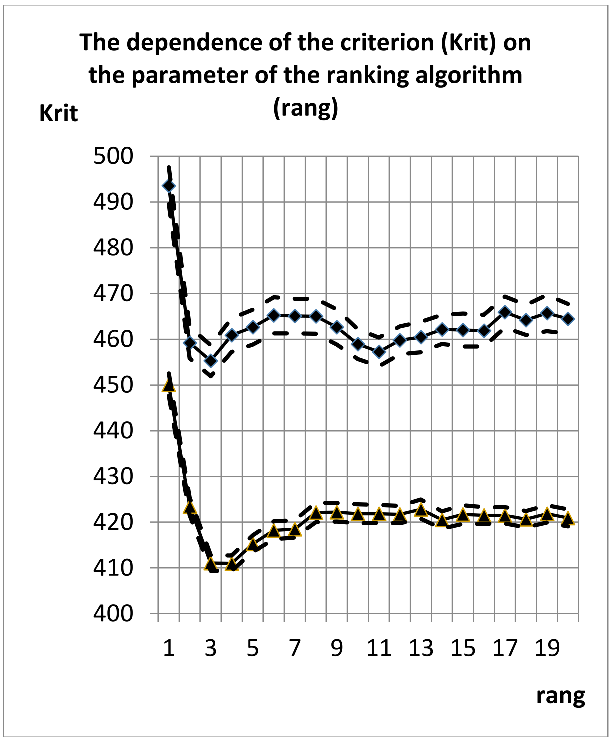

Fig 5. Determination of the optimal parameters parameter of the ranking for the modification of the ant colony method.

But the algorithm coped well with the non-optimal evaporation coefficient, showing more than 95% success at values greater than 0.25. But this success led to a sharp increase in the number of iterations of the ant colony method, and, consequently, in the search time for a solution. The optimal parameter of the evaporation coefficient remains a value from the range (0.8; 0.95), with them the graph is simply not reset. For all graphs, resetting the decision graph in the absence of new found solutions at the iteration (light gray triangles) shows the best results.

It was calculated that the algorithm for resetting the decision graph turned out to be optimal for the case when no new solutions were found. After resetting the decision graph, it is possible to establish its rational state by adding a pheromone from the rational routes found so far. This approach allows us to consider many solutions around the already found rational solutions

The main problem of the ant colonies algorithm, graph looping (the number of successful runs does not exceed 0.8 for any number of iterations that determine looping), both algorithms successfully solve (Fig. 6). Already with the number of iterations equal to 1750, the percentage of successful runs exceeds 95% in both algorithms. But this is associated with an increase in the number of resets of the decision graph and, as a result, an increase in the number of iterations of the algorithm and the number of solutions considered, which means an increase in the search time. Due to the different behavior of the algorithm as a result of the reset, the confidence intervals of the estimate of the mathematical expectation are large, i.e. the iterations of the algorithm are very different. In this case, the graph is reset more often if the number of solutions is chosen as a criterion, which leads to the best indicators in terms of the percentage of successful runs. When using the algorithm for stopping the run when the constraints are reached, the use of resetting the decision graph also improves the probability of finding a solution and increases the time of this search.

It should be noted that the execution time of each iteration of the algorithm in one run is not the same. (Fig. 7). This is due to the need to store all solutions and search in this set for the solution found by the agent. This feature is necessary for problems in which the time for calculating the values of the system's criteria for a certain solution is much longer than the running time of the ant colony method, for example, if the simulation model is involved in calculating the criteria. In our case, the mathematical operations for calculating the total time are quite simple and it is possible not to store the entire array of solutions, which will lead to a sharp acceleration of the operation of the algorithm for problems requiring consideration of a large number of solutions. But, in this case, it is difficult to use the algorithm for resetting the graph by the number of solutions.

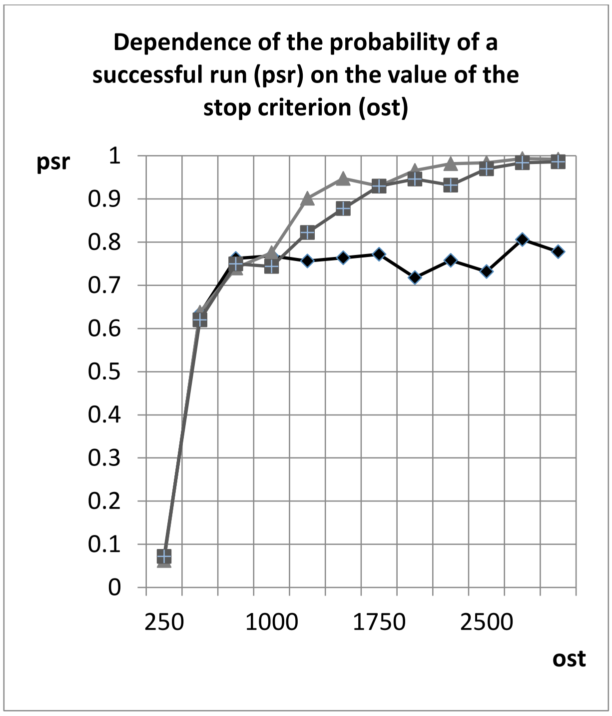

Fig 6. Consideration of the influence of resetting the decision graph of a successful run algorithm.

As you can see from the graphs, the best value of the number of input solutions when resetting the graph varies from 2 to 5. With such values, the maximum number of successful runs is ensured, the histogram. It should be noted that almost all values, except for very high 20 and more, provide a greater number of successful iterations than in the absence of entering solutions on the graph (when the value is equal to 0). A "successful" run is one that in a limited amount of time was able to find a solution that satisfies very tight constraints.

The number of iterations decreases when the number of inputted solutions changes (black graph), since inputting a rational distribution of weights on the graph leads to a faster search for a solution that satisfies the constraints. The dashed lines mark the confidence interval. The number of considered solutions also decreases. But the number of graph resets increases when the number of input solutions increases to 6. This is followed by a constant value of the number of discharges of the graph with its subsequent decrease. This is due to the fact that the introduced solutions affect the initial movement of the agents; as a result, most agents follow the same routes, which lead to rare drops of the graph.

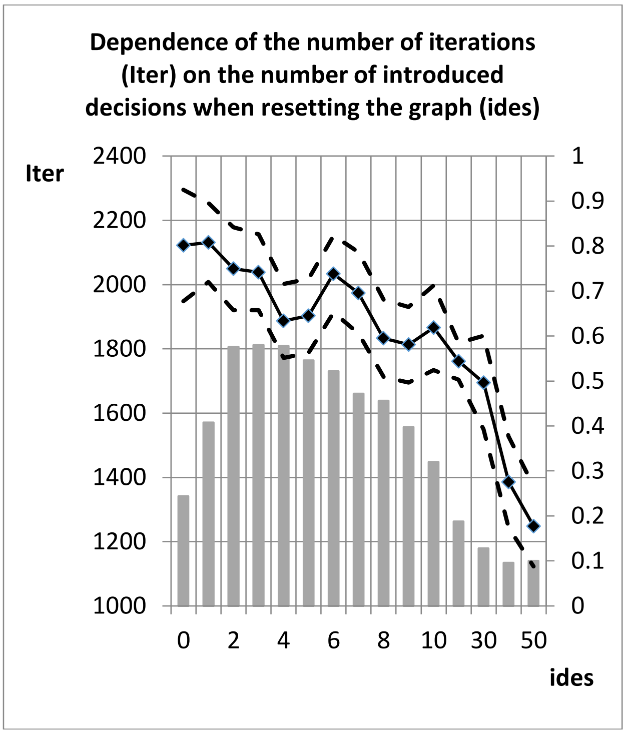

Fig 7. Consideration of introduced decisions when resetting the decision graph on the results of the algorithm.

In general, an endless search for a solution that satisfies the constraints can be carried out by setting the hang criterion very large. If unfeasible limits are set, then you can track the best solution, which will gradually improve.

VI. Conclusion

- An approach has been proposed that makes it possible to estimate the execution time of a work divided into tasks. Within the framework of this approach, for each task, you can set the execution time in the form of a fuzzy set. At the same time, it was taken into account that fuzzy sets are set by the employees themselves and the total time for completing a task is determined by the generalized time for completing the task by all employees. This takes into account the time required for employee interaction.
- An algorithm for generalizing fuzzy sets given on various bearing sets is presented. The procedure for calculating the generalized time of the task execution is described by passing to the employee's productivity.

- A modification of the ant colony method for solving the problem of decision support is proposed. For this modification, the agents move along a special decision graph. The algorithm of this modification and its features are described. This modification was used to solve the problem of assigning employees to tasks.
- Rational parameters of the ant colony method modification were determined. These parameters allow you to speed up the time to find rational solutions.
- The main problems of the ant colony method associated with looping the algorithm on one solution are described. To solve this problem, it is proposed to use the reset of the decision graph. As a result, it becomes possible not to be limited by some criterion for stopping the ant colony method, since the improvement of the solution occurs continuously.

ACKNOWLEDGMENT

This study was financed by a grant from the Plekhanov Russian University of Economics

REFERENCES

[1] E. V. Jamai, A.S. Zinchenko "Cost management of a mechanical engineering enterprise in modern financial conditions" Socio-economic and humanitarian research. Vol. 7. pp. 110-113. 2015 (in Russian)

[2] A Guide to the Project Management Body of Knowledge (PMBOK® Guide) - Fifth Edition.

[3] N. Bushuyeva, D. Bushuiev, V. Bushuieva "AGILE Leadership of Managing Innovation Projects" Innovate Technologies and Scientific Solutions for Industries, Vol 4(10), 2019, DOI: 10.30837 / 2522-9818.2019.10.077

[4] Koçyiğit Yeşim, Akkaya Bulent. "The Role of Organizational Flexibility in Organizational Agility: A Research on SMEs" Business Management and Strategy, Vol 11(1), pp. 110-123, 2020 DOI: 10.5296 / bms.v11i1.16867

[5] Manifesto for Agile Software Development URL: http://agilemanifesto.org/ (Accessed 30 September 2020).

[6] S.P. Chen, Y.J. Hsueh "A Simple Approach to Fuzzy Critical Path Analysis in Project Networks" Applied Mathematical Modeling, vol. 32, pp. 1289-1297, 2008 DOI: 10.1016 / j.apm.2007.04.009

[7] S.T. Liu, "Fuzzy Activity Times in Critical Path and Project Crashing Problems" Cybernetics and Systems: An international Journal, vol. 34, pp. 161-172, 2003 DOI: 10.1080 / 01969720302865

[8] Popescu Costin-Ciprian. "On Critical Path with Fuzzy Weights" Economic Computation and Economic Cybernetics Studies and Research, vol. 52 (4), 2018 DOI: 10.24818/18423264/52.4.18.04

[9] P.C. Shih, "Analysis of critical paths in a project network with fuzzy activity times" European Journal of Operational Research, vol. 183 (1), pp. 442-459, 2007, DOI: 10.1016 / j.ejor.2006.06.053

[10] Z. Pawe, "On computing the latest starting times and floats of activities in a network with imprecise durations" Fuzzy Sets and Systems, vol. 150 (1), pp. 53-76, 2005, DOI: 10.1016/j.fss.2004.08.007

[11] S.H. Nasution, "Fuzzy critical path method" IEEE Trans Systems Man Cybernet, vol. 24,pp. 48-57, 1994, DOI: 10.1109/21.259685

[12] T.C. Chen, F.H. Sue. "Applying fuzzy method for measuring criticality in project network" Information Sciences, vol. 177(12), pp. 2448-2458, 2007, DOI: 10.1016 / j.ins.2007.01.035

[13] A.A. Zatsarinny, V.V. Korotkov, M.G. Matveev "Modeling the processes of network planning of a portfolio of projects with heterogeneous resources in the conditions of fuzzy information" Informatics and its application, Vol. 13, Issue. 2, p. 92-99, 2019. (in Russian)

[14] V.A. Sudakov, Yu.P. Titov "Solving the problem of determining the time of work by a group of employees using fuzzy sets" Open Education, Vol 23(5), pp. 74-82, 2019, (in Russian) DOI: 10.21686 / 1818-4243-2019-5-74-82

[15] V.A. Sudakov, Yu.P. Titov, "Calculation of time to complete a task by a group of employees using fuzzy sets for the sustainable development of the enterprise" Entrepreneurship and Sustainability Vol. 8 (1), pp. 1001-1011, 2020 http://doi.org/10.9770/jesi.2020.8.1(67)

[16] A. Colorni, M. Dorigo, V. Maniezzo. "Distributed optimization by ant colonies" Proceedings of the First European Conference on Artificial Life, ECAL'91. Elsevier, Paris, France. pp. 34-142, 1992

[17] A.P. Karpenko, K.A. Chernobrivchenko "Optimization efficiency by the method of continuously interacting ant colony (CIAC)" Science and education. Electronic journal, vol. 2, 2011, Open access http://technomag.edu.ru/ (in Russian), (Accessed 30 September 2020).

[18] S.D. Shtovba "Ant algorithms" Exponenta Pro, Mathematics in applications, vol. 4 (4), pp. 70-75, 2003, (in Russian)

[19] V.P. Karelin "Models and methods of graph theory in decision support systems" Bulletin of the Taganrog Institute of Management and Economics, vol. 2(20), pp. 69-73, (in Russian)

[20] Yu.P. Titov, E.A. Davydkina "Expanding the capabilities of the ant colonies method by using fuzzy sets" Trends in the development of science and education, Vol 2(54), pp. 16-19, 2019, (in Russian)

[21] V.A. Sudakov, Yu.P.Titov "Application of the modified ant colony method to find the rational assignment of employees to tasks using fuzzy sets" Statistics and Economics. Vol. 17 (3), pp. 79-91. 2020 (in Russian) DOI: 10.21686 / 2500-3925-2020-3-79-91