OPTIMIZATION OF A-STAR SEARCH ALGORITHM

D.S. Piskorskii, *piskorskiids@susu.ru*,
F.K. Abdullin, *abdullinfk@susu.ru*,
A.R. Nikolaeva, *komogorovaar@susu.ru*
South Ural State University, Chelyabinsk, Russian Federation

Introduction. Autonomous mobile robots must be able to plan global and local motion paths. The A-star path planning algorithm allows us to calculate the shortest path between the starting and end points on a map with known static obstacles. In real conditions, when additional information about the area is entered (difficult or dangerous sections, areas with speed limits) and the cost of overcoming them is taken into account, A-star can lead to a non-optimal, for these conditions, solution of the problem. **Aim.** Consider options for optimizing the A-star path planning algorithm for use in various conditions with restrictions on the number of turns, linking to critical points on a map of the area, difficult and dangerous areas and assess the quality of the optimization. **Materials and methods.** Research is carried out by computer simulation of the A-star algorithm and options for its optimization in the MATLAB environment. The criteria for evaluating the quality of optimization are focused primarily on computational time and the path optimality with respect to the selected parameters. **Results.** The results of path calculation performed using the A-star algorithm before and after optimization are presented. In both cases, the following are estimated and compared: calculation time, number of analyzed polygons, number of turns and path length. **Conclusion.** In most cases, the optimization of the algorithm increases the path length and calculation time, but not significantly. Moreover, the new path corresponds to the given conditions, is the shortest in these conditions and, therefore, is optimal. The considered optimization options allow you to calculate the path taking into account additional information, estimate the path length and computational time. On the basis of these evaluations, it is possible to choose path planning method suitable for individual scenario.

Keywords: path planning, A-star algorithm, motion path optimization, mobile robotic, path cost, model.

Introduction

Today, mobile robots are engaged in various tasks related to the delivery of goods, reconnaissance, automated patrol, and search and rescue operations. Autonomous mobile robots must be able to plan global and local motion paths, create a three-dimensional model of the environment, determine its location in space and control actuators to keep motion along the planned path [1].

The path planning is an important and non-trivial task. Therefore a lot of studies are currently underway in this area. Various algorithms are being developed for calculating the path both in a static environment with known obstacles and in unknown and dynamic environments where it is impossible to have prior complete information that can be given to the robots before operation [2–8].

One of the most effective path finding algorithms in a static environment, widely used in practice, is the A-star algorithm [9, 10]. It is a modernized version of Dijkstra's algorithm. A-star allows us to calculate the path at the lowest cost from the initial peak to the final in a weighted directed graph. The least path cost in the algorithm means the shortest distance between the selected points.

However, if we are dealing with real objects that have some limitations (kinematic, controls, etc.) or when introducing additional information about the map of the area and taking into account the costs of overcoming them (the cost of maneuvering, overcoming hills, etc.) – the considered algorithm can lead to a non-optimal solution of the problem for given conditions [11, 12]. Therefore, for practical application, the algorithm must be optimized for real conditions including a number of restrictions [13]: kinematic restrictions (maximum speed, minimum turning radius, etc.), control limitations (remote control along the motion path), restrictions on the degree of danger of the chosen path and others.

1. Implementation of A-star algorithm in MATLAB

For research, the A-star algorithm was implemented as a computer model in the MATLAB modeling environment [14, 15]. The model's interface is a 20×20 polygon field where the starting point "Start", the end point "Finish" are marked and obstacles are set (Fig. 1).
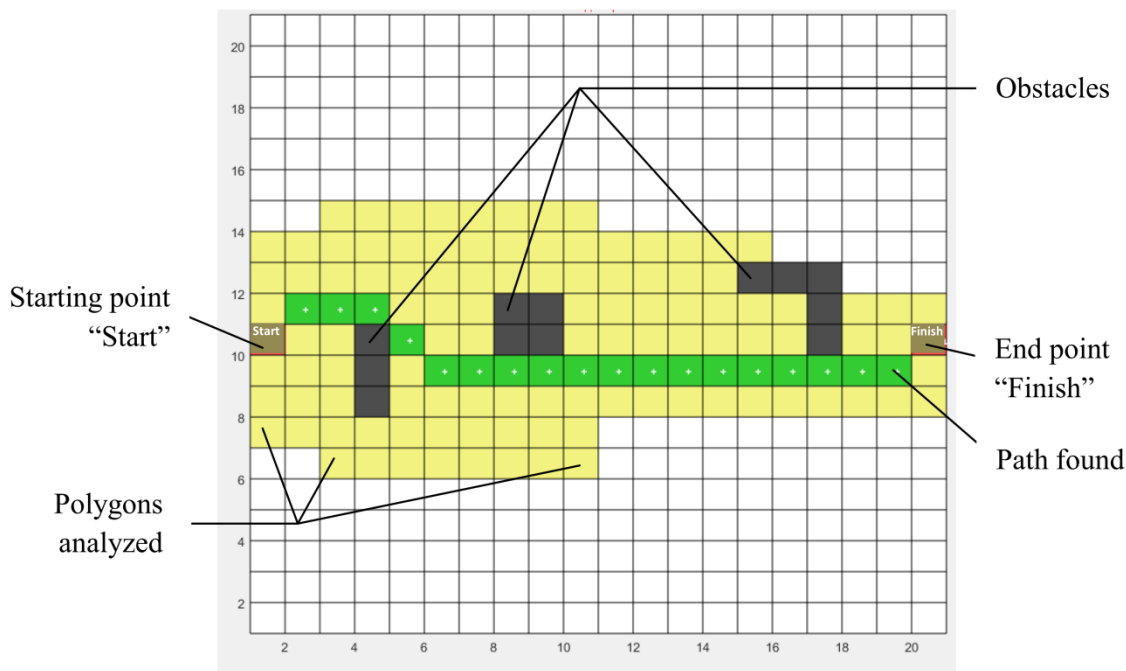
Computational time, ms	31.3
Polygons analyzed	126
Path length (in polygons)	19
Number of turns	3

Fig. 1. Example of path calculation

The path calculation algorithm works as follows.

1. In the first step, two lists of polygons are created: a) pending analysis, b) analyzed. A starting point is added to the list of pending points.

2. For all neighboring polygons that are no obstacles, the coefficient F_n is calculated:

$$F_n = G_n + H_n, \tag{1}$$

where G_n is the cost of transitions from "Start" to the current point, H_n – the remaining distance from the current point to "Finish", and the number of the parent polygon from which the transition was made is remembered.

3. All enumerated polygons are recorded in the list of pending analysis. Next, from this list, the polygon $P_{\min Fn}$ with the lowest value of F_n is selected.

If $P_{\min Fn}$ is the final polygon, then the route is found and the algorithm is finished. If not, $P_{\min Fn}$ is moved from the pending analysis list to the analyzed list.

4. Then, for each of the P_i ranges adjacent to it, with the exception of obstacles, the conditions are checked:

a) if P_i is included in the list of analyzed polygons, then skip the calculation, if not, go to point b);

b) if P_i is not included in the list of pending analysis, then we add it there by calculating F_{ni} and remembering the link to the parental polygon $P_{\min Fn}$, if P_i is included in the list of pending analysis, then compare the current F_{ni} with the resultant F_{nj} for this polygon. If $F_{ni} < F_{nj}$, then a shorter less expensive, path to the current polygon was found, therefore, replace F_{nj} with F_{ni}.

5. Repeat steps 2 to 4 until the end point of the route is reached.

6. If the list of fields awaiting the analysis is empty and the endpoint is not reached, then the route does not exist.

2. Optimization of algorithm for the number of turns

A-star is optimal in terms of finding a path with a minimum length. The algorithm is modified by adding optimality to the minimum number of turns in the path. For this purpose additional cost coefficients W_n that increase the cost of the path when turning are introduced. In this case, the formula for calculating the cost of the path will be:

$$F_n = G_n + H_n + W_n. \tag{2}$$

The simulation results without optimization are shown in Fig. 2a, with optimization in Fig. 2b.
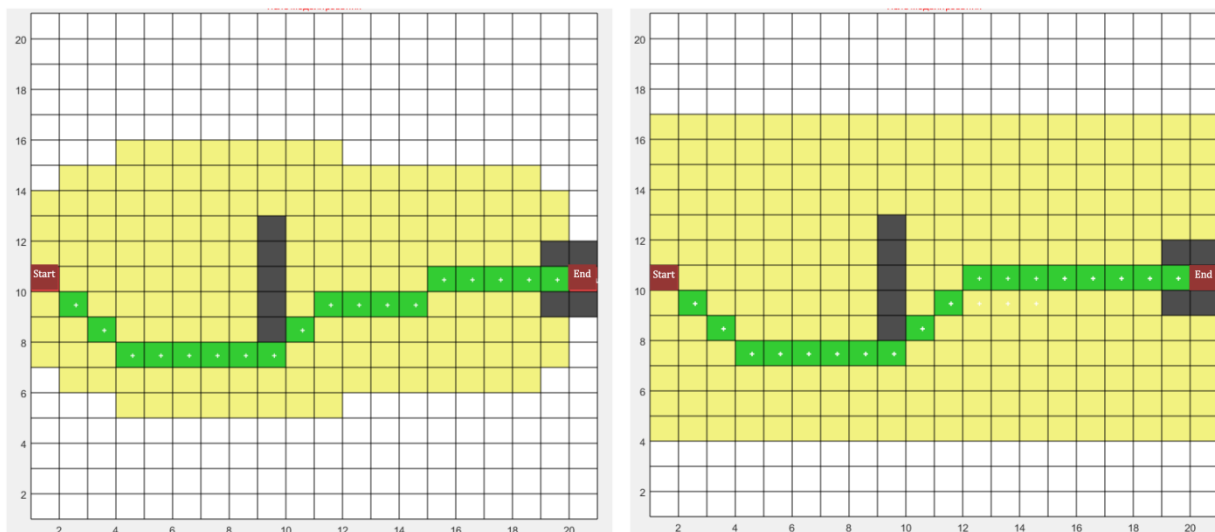
Computational time, ms	46.0		Computational time, ms	62.5
Polygons analyzed	177		Polygons analyzed	251
Path length (in polygons)	19		Path length (in polygons)	19
Number of turns	5		Number of turns	3

<div style="text-align:center">a) results without optimization   b) turn-optimized results</div>

Fig. 2. Calculation of the optimal path

The results obtained show that, after optimization, the number of turns was reduced from five to three, while the path length, measured in the number of passed polygons, did not change. At the same time, the number of analyzed polygons increased and, as a result, so did the computational time, but not critically.

3. Optimization by complexity or time of the path

This type of optimization is carried out by introducing weighting factors that increase or decrease the cost of passage of the indicated polygons in the field. In practice, the increase in the cost of crossing the polygon corresponds to the complexity of the path along this route (mountainous or marshy terrain, or dangerous area) or the time taken to overcome the path (busy track). Increasing the cost of passage through such areas will allow us to find another way that these areas will avoid, the more the higher the cost of their passage.

Fig. 3b shows the results of a path search in the case when the cost of passing the polygons located in the lower part of the modeling field is doubled compared to Fig. 3a.

As a result of the simulation, the path along the upper part of the field was calculated for polygons with less cost. At the same time, the number of analyzed polygons decreased and, as a result, the calculation time.
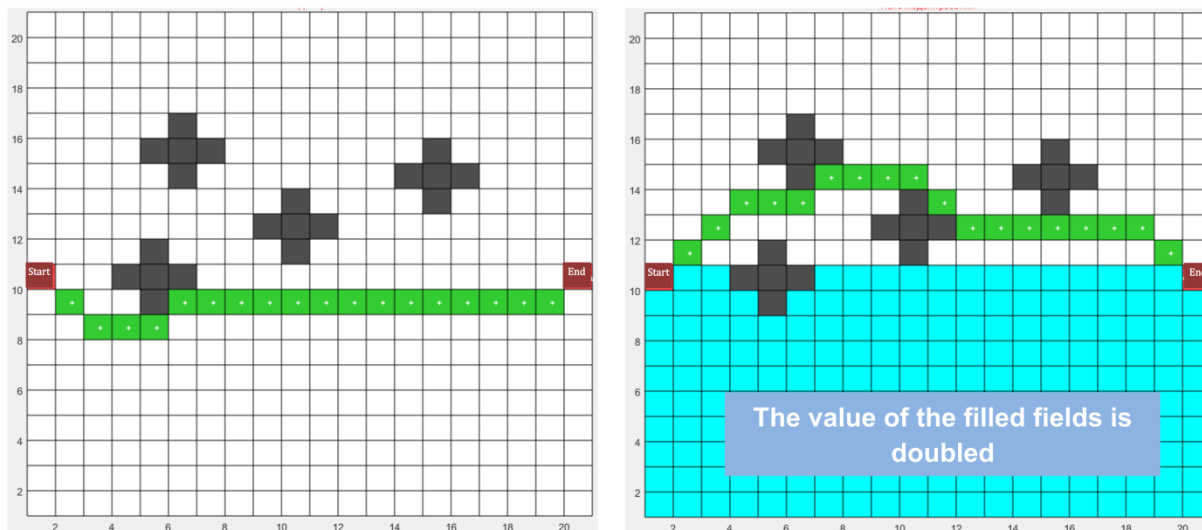
Computational time, ms	15.6	Computational time, ms	14.8
Polygons analyzed	124	Polygons analyzed	97
Path length (in polygons)	19	Path length (in polygons)	19
Number of turns	3	Number of turns	7

a) results without optimization   b) turn-optimized results

Fig. 3. Calculation of the optimal path

4. Algorithm optimization based on critical points

The inverse solution of the previous problem can be used in cases where it is necessary for the path of the mobile robot to passes through certain areas on the map, even if the path increases. This may be suitable if the mobile robot must constantly keep in touch with control centers or collect information through relay stations installed in predetermined positions. In this case, when planning the path, it is necessary for the mobile robot to be in the coverage area of these stations (Fig. 4).
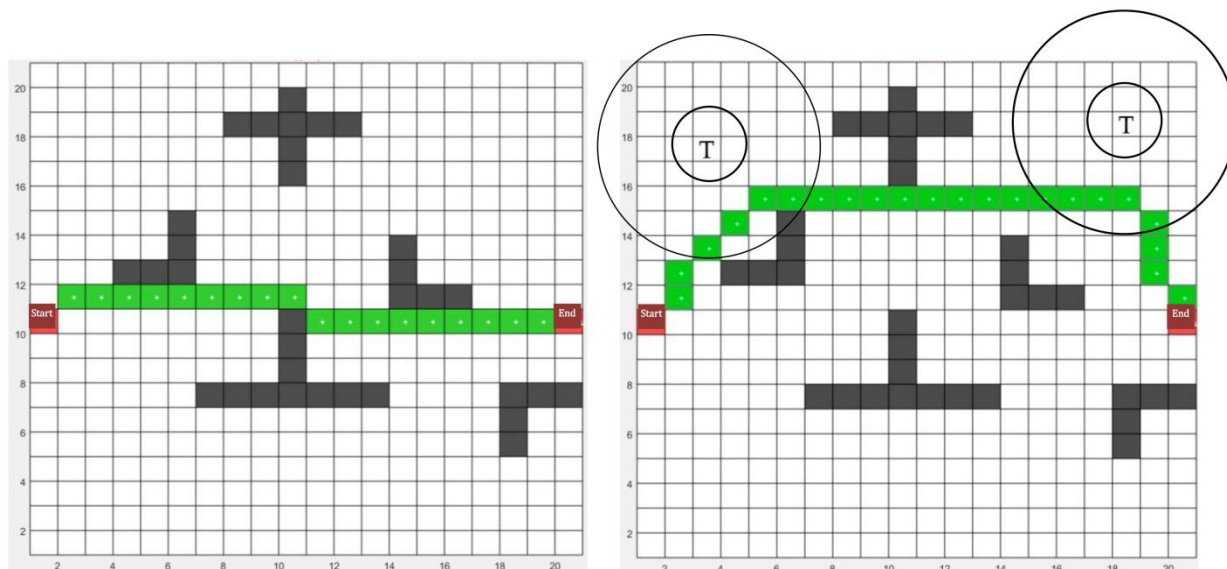
Computational time, ms	10,7	Computational time, ms	16,8
Polygons analyzed	76	Polygons analyzed	124
Path length (in polygons)	19	Path length (in polygons)	23

a) results without optimization   b) turn-optimized results

Fig. 4. Calculation of the optimal path

This can be achieved by adjusting weights. That is by significant increase in weights in the areas of the map (remote from the station by a distance exceeding the range of radio communications with a mobile robot).

Fig. 4 shows the calculations of the optimal path in the presence of two relay stations indicated on the modeling field by the symbol "T". As a result of the calculation, we obtained a path (Fig. 4b) that deviates from the calculated shortest path (shown in Fig. 4a), however, this path passes through the polygons which are least distant from those allocated with radio coverage and therefore being more optimal for these conditions.

Conclusion

As a result of the studies, options for optimizing the A-star path planning algorithm for its application in conditions with a limited number of turns, linking to critical points on a map of the area, as well as difficult or dangerous sections are proposed. In most cases, as a result of optimization, the path length and its computational time increase, but not significantly. Moreover, the new path corresponds to the given conditions, is the shortest for these conditions and, therefore, is optimal. The developed model allows us to estimate the time and necessary computing resources spent on calculating the path. On the basis of these evaluations, it is possible to choose path planning method suitable for individual scenario.

Of further interest is the task of optimizing the algorithm with incomplete reliability of a priori information or its absence in some parts of the analyzed terrain map.

References

1. Noskov V.P., Rubtsov V.I., Rubtsov I.V. *Matematicheskie modeli dvizheniya i sistemy tekhnicheskogo zreniya mobil'nykh robototekhnicheskikh kompleksov. Uchebnoe posobie* [Mathematical Models of Motion and Systems of Technical Vision of Mobile Robotic Complexes]. Moscow, 2015, 94 p.

2. Koenig S., Likhachev M., Furcy D. Lifelong Planning A*. *Artificial Intelligence*, 2004, vol. 155, no. 1, pp. 93–146. DOI: 10.1016/j.artint.2003.12.001

3. Kilibarda G., Kudryavtsev V.B., Ushchyumlich Sh. [The Independent Systems of Automata in the Labyrinth]. *Discrete Mathematics*, 2003, vol. 15, iss. 2, pp. 3–39. (in Russ.)

4. Maksimova E.I. [Comparison of the Quality of the Results of the A-Star Algorithm and its Modifications for the Road Network when Choosing a Route, Taking into Account the Direction of Movement at the Intersection]. *Siberian Science Bulletin*, 2014, no. 4, pp. 117–123. (in Russ.)

5. Dan B. Marghitu. *Mechanisms and Robots Analysis with MATLAB*. Springer-Verlag London Limited, 2009, 479 p. DOI 10.1007/978-1-84800-391-0

6. Zeng W., Church R.L. Finding Shortest Paths on Real Road Networks: the Case for A*. *International Journal of Geographical Information Science*, 2009, vol. 23, no. 4, pp. 531–543. DOI: 10.1080/13658810801949850

7. Lei T., Songyi D., Gangxu G., Kunli Zh. A Novel Potential Field Method for Obstacle Avoidance and Path Planning of Mobile Robot. *3rd IEEE International Conference Computer Science and Information Technology*, 2010, vol. 9, 6 p. DOI: 10.1109/ICCSIT.2010.5565069

8. Choset H., Lynch K., Hutchinson S., Kantor G., Burgard W., Kavraki L., Thrun S. *Principles of Robot Motion: Theory, Algorithms, and Implementations*. MIT Press, 2005, 603 p.

9. Liu V. Methods of Path Planning in an Environment with Obstacles (Review). *Mathematics and Mathematical Modeling*, 2018, no. 1, pp. 15–58. (in Russ.) DOI: 10.24108/mathm.0118.0000098

10. Lavrenova P.O., Afanasyeva I.M., Magid E.A. [Route Planning for an Unmanned Ground Robot Taking into Account Many Optimization Criteria]. *Results of Scientific-Practical Seminar "Unmanned Vehicles with Elements of Artificial Intelligence"*, 2015, pp. 10–20. (in Russ.)

11. Alonzo Kelly. *Mobile Robotics. Mathematics, Models and Methods*. Cambridge University, 2013, 716 p. DOI: 10.1017/CBO9781139381284

12. Spyros G. Tzafestas. *Introduction to Mobile Robot Control*. School of Electrical and Computer Engineering National Technical University of Athens, 2014, 691 p. DOI: 10.1016/B978-0-12-417049-0.00004-3

13. Wallgrun J.O. Voronoi Graph Matching for Robot Localization and Mapping. *Transactions on Computational Science IX*. Springer, 2010, pp. 76–108. DOI: 10.1007/978-3-642-16007-3_4

14. Gonzalez R., Mahulea C. and Kloetzer M. A Matlab-Based Interactive Simulator for Teaching Mobile Robotics. *IEEE 2015: Int. Conf. on Autom. Science and Engineering*, 2015, pp. 310–315. DOI: 10.1109/CoASE.2015.7294097

15. Goldshtein A.L. *Optimizatsiya v srede MATLAB: ucheb. posobie* [Optimization in MATLAB: Textbook]. Perm, Publishing Perm National Research Polytechnic University, 2015, 192 p.

_____

ОПТИМИЗАЦИЯ АЛГОРИТМА ПЛАНИРОВАНИЯ ПУТИ A-STAR

Д.С. Пискорский, Ф.Х. Абдуллин, А.Р. Николаева
Южно-Уральский государственный университет, г. Челябинск, Россия

Введение. Автономные мобильные роботы должны уметь самостоятельно планировать глобальную и локальную траектории своего движения. Алгоритм планирования пути A-star позволяет вычислить кратчайший путь между начальной и конечной точками на карте местности с известными статичными препятствиями. В реальных условиях при введении дополнительной информации о карте местности (труднопроходимые или опасные участки, участки с ограничением скорости) и учете затрат на их преодоление алгоритм может приводить к неоптимальному для данных условий решению задачи. **Цель исследования.** Рассмотреть варианты оптимизации алгоритма планирования пути A-star для применения в различных условиях, имеющих ограничения по количеству поворотов, привязку к критическим точкам на карте местности, труднопроходимые и опасные участки. Оценить качество проведенной оптимизации. **Материалы и методы.** Исследования проводятся путем компьютерного моделирования алгоритма A-star и вариантов его оптимизации в среде MATLAB. Критериями оценки качества оптимизации алгоритма являются скорость расчета пути и его оптимальность относительно выбранных параметров. **Результаты.** Приводятся результаты расчета пути, выполненные с помощью алгоритма A-star до и после оптимизации. В обоих случаях оцениваются и сравниваются: время расчета, количество проанализированных полигонов, число поворотов и длина пути. **Заключение.** В большинстве случаях в результате оптимизации алгоритма увеличивается длина пути и время расчета, но незначительно. При этом новый путь соответствует заданным условиям, является кратчайшим в этих условиях и, следовательно, оптимальным. Предложенные в статье варианты оптимизации позволяют вычислить путь с учетом дополнительной информации, оценить длину пути и скорость расчета. На основе этих оценок можно выбрать метод планирования пути, подходящий для отдельного сценария.

Ключевые слова: планирование пути, алгоритм A-star, оптимизация траектории движения, мобильный робот, стоимость пути.

Литература
1. Носков, В.П. Математические модели движения и системы технического зрения мобильных робототехнических комплексов: учеб. пособие / В.П. Носков, В.И. Рубцов, И.В. Рубцов. – М.: Изд-во МГТУ им. Н.Э. Баумана, 2015. – 94 с.

2. Koenig, S. Lifelong planning A* / S. Koenig, M. Likhachev, D. Furcy // Artificial Intelligence. – 2004. – Vol. 155, no. 1. – P. 93–146. DOI: 10.1016/j.artint.2003.12.001

3. Килибарда, Г. Независимые системы автоматов в лабиринте / Г. Килибарда, В.Б. Кудрявцев, Ш. Ушчюмлич // Дискретная математика. – 2003. – Т. 15, вып. 2. – С. 3–39.

4. Максимова, Е.И. Сравнение качества результатов алгоритма «A-star» и его модификации для дорожной сети при выборе маршрута с учетом направления движения на перекрестке / Е.И. Максимова // Вестник науки Сибири. – 2014. – № 4. – С. 117–123.

Краткие сообщения

5. Dan B. Marghitu. Mechanisms and Robots Analysis with MATLAB / Dan B. Marghitu. – Springer-Verlag London Limited, 2009. DOI 10.1007/978-1-84800-391-0

6. Zeng, W. Finding shortest paths on real road networks: the case for A* / W. Zeng, R.L. Church // International Journal of Geographical Information Science. – 2009. – Vol. 23, no. 4. – P. 531–543. DOI: 10.1080/13658810801949850

7. A novel potential field method for obstacle avoidance and path planning of mobile robot / T. Lei, D. Songyi, G. Gangxu, Zh. Kunli // 3-rd IEEE International Conference Computer Science and Information Technology. – 2010. – Vol. 9. – 6 p. DOI: 10.1109/ICCSIT.2010.5565069

8. Principles of Robot Motion: Theory, Algorithms, and Implementations / H. Choset, K. Lynch, S. Hutchinson et al. – MIT Press, 2005. – 603 p.

9. Лю В. Методы планирования пути в среде с препятствиями (обзор) // Математика и математическое моделирование. – 2018. – № 1. – С. 15–58. DOI: 10.24108/mathm.0118.0000098

10. Лавренова, Р.О. Планирование маршрута для беспилотного наземного робота с учетом множества критериев оптимизации / Р.О. Лавренова, И.М. Афанасьева, Е.А. Магид // Результаты всероссийского научно-практического семинара «Беспилотные транспортные средства с элементами искусственного интеллекта». – 2015. – С. 10–20.

11. Kelly, Alonzo. Mobile Robotics. Mathematics, Models and Methods / Alonzo Kelly. – Cambridge university, 2013. – 716 p. DOI: 10.1017/CBO9781139381284

12. Tzafestas, Spyros G. Introduction to Mobile Robot Control / Spyros G. Tzafestas. – School of Electrical and Computer Engineering National Technical University of Athens, 2014. – 691 p. DOI: 10.1016/B978-0-12-417049-0.00004-3

13. Wallgrun J.O. Voronoi graph matching for robot localization and mapping / J.O. Wallgrun // Transactions on computational science IX. – Springer, 2010. – P. 76–108. DOI: 10.1007/978-3-642-16007-3_4

14. Gonzalez, R. A Matlab-Based Interactive Simulator for Teaching Mobile Robotics / R. Gonzalez, C. Mahulea, M. Kloetzer // IEEE CASE'2015: Int. Conf. on Autom. Science and Engineering. – 2015, pp. 310–315. DOI: 10.1109/CoASE.2015.7294097

15. Гольдштейн, А.Л. Оптимизация в среде MATLAB: учеб. пособие / А.Л. Гольдштейн. – Пермь: Изд-во Перм. нац. исследоват. политехн. ун-та, 2015. – 192 с.

Пискорский Дмитрий Сергеевич, аспирант, старший преподаватель кафедры инфокоммуникационных технологий, Южно-Уральский государственный университет, г. Челябинск; piskorskiids@susu.ru.

Абдуллин Фаиль Хамидуллович, старший преподаватель кафедры инфокоммуникационных технологий, Южно-Уральский государственный университет, г. Челябинск; abdullinfk@susu.ru.

Николаева Алиса Робертовна, аспирант, инженер кафедры инфокоммуникационных технологий, Южно-Уральский государственный университет, г. Челябинск; komogorovaar@susu.ru.