Collision-Free Path Planning Algorithm for Group of Robots in Spatio-Situational Uncertainty

Dmitrii Motorin, Serge Popov, Vladimir Muliukha
Peter the Great St. Petersburg Polytechnic University St. Petersburg, Russia
d.e.motorin@gmail.com, popovserge@spbstu.ru, vladimir@mail.neva.ru

***Abstract*—In the process of robots paths planning potential collisions of trajectories occur which can be identified and avoided. In case of spatio-situational uncertainty, collision solution can be repeated, with collisions appearing and disappearing. In highly dynamic environment, the more time past since the last collision the lower probability of its re-occurrence. The paper presents the study of the minimum range of the individual robot collision avoidance for a group of robots under condition of spatio-situational uncertainty. The results have been applied to develop an adaptive collision avoidance algorithm. The model of environment and movement of robots are used to determine the range of the collision avoidance. The collisions repeating from step to step, allowed calculating the range of the collision avoidance with respect to environmental change frequencies. The adaptive algorithm for collision avoidance was designed to research effectiveness of the selected ranges. Authors also compared the systems without algorithm, with adaptive range and with fixed coefficients. The effectiveness/efficiency was evaluated by comparing results of modeling two systems without an adaptive algorithm and with it by call of the collision avoidance algorithm.**

I. Introduction

When robots' paths are planned and when the robots move their trajectories can cross. Let's assume such crossings refer to as collisions. Collision avoidance is carried out by adjusting the trajectories of one or several robots. The movement of a large group of robots in a limited space involves finding solutions to a great number of such problems along all trajectories.

While in static environment calculations are conducted once, at the initial stage of planning (the stage of planning, before movement), in dynamic environment robots recalculate trajectories every time when changes affecting the path occur. In case of high spatio-situational uncertainty several collision resolution would be useless because probability of the collision occurrence decreases as the distance from the current position to the spot of the potential collision increases. If low-probability collisions are excluded from the calculations the time of path planning is considerably reduced.

The main issue is to determine the number of collision avoidance for each robot, depending on the environment dynamics. The results will allow us to build an adaptive control algorithm for collision avoidance of dynamic map frequency. It will significantly reduce the requirement to calculation paths for group of the robots.

II. Related works

The task of collision avoidance has a number of approaches and solutions. For example, [1] present the approach based on separating space according to geometrical and topological principles on non-overlapping areas, in which large groups of robots move avoiding collisions by following traffic rules. The environment is determinate and does not change over time.

Work [2] introduce the concept of the super-graph, which is a combination of simple trajectories for each robot for collision avoidance. The trajectory of each robot consists of a combination of basic movement patterns. The sum of all subgraphs amounts to the super-graph. Paths planning problem solves by computing coordinated retractions.

Solution to the collision avoidance problem in static environment without obstacles was discussed in [3]. The authors considered a centralized approach to the coordination of a group of robots. Collision avoidance occurs due to changes in the robots' speed and the deviation of their trajectories. A similar method of collision avoidance is used in [4] with the main difference consisting in decentralization of the algorithm. A robot tries to avoid the closest possible collision with minimal distances to objects being under control to the object.

In [5] consider movement of the group of robots by using a set of predetermined individual points in empty space. Lyapunov-like barrier functions is used to avoid collisions.

The environmental map is divided into the grid in [6]. Collision avoidance is described by using grid method with time windows and the updating rule of pheromone in ant colony algorithm.

Collision avoidance considered in [7] employs time delays in the initial stages of movement on a static known map.

Collision avoidance by using control barrier method is presented in [8]. With minimum distance between robots being defined parametrically.

All works presented above consider motion only in a static environment while in real tasks groups of robots often act in a dynamic environment. For example, [9] analyze not only robot trajectories but also the motion of obstacles. Path planning algorithm minimizes the trajectory length and the collision probability. In [10] the dynamics of the environment is provided by replacing obstacles. The robot trajectory to the target is adjusted during movement as a reaction to the approaching obstacle.

In the above-mentioned works the movement of robots occurs mostly in the obstacle-free or static environment, which limits the applications of these algorithms in conditions of high spatio-situational uncertainty, such as: vehicles autonomous movement on rough terrain, group of robots movement in urban environment or a group of space robots movement on celestial bodies, where environmental changes are of probabilistic nature.

III. Problem Statement

The purpose of the work is to research and develop of collision-free path planning adaptive algorithm for group of robots in conditions of spatio-situational uncertainty.

The problem is considered on a dynamic map E marked up with a regular grid, which consists of passable areas E^P and impassable ones, filled with obstacles, E^W. The portion of the latter is $\alpha\%$. When robots move the position of obstacles on te map change by $\omega\%$. with every step.

The initial position of n robots on the map is fixed $A = \{A_1, \ldots, A_k, \ldots, A_n\}$. All robots are grouped together $R = \{R_1, \ldots, R_k, \ldots R_n\}$. Target points $Z = \{Z_1, \ldots, Z_k, \ldots, Z_n\}$ in which robots have to arrive are arranged on the map. Each robot should arrive at one of the points. Robots choose their destination points at the beginning of movement randomly. To archive the goal robots plan their trajectories $L = \{L_1, \ldots, L_k, \ldots L_n\}$:

$$L_k = \{A_{Rk}, \ldots, l_i, \ldots, Z_{Rk}\} \in L : l_i \in E^P.$$

The criterion of the quality of the developed adaptation algorithm:

$$\sum cs_i \to \min \& \sum cf_i \to \min,$$

where cs – number of authorized collisions, cf – number of collision search function calls.

IV. Algorithm

The map E is consist of a regular grid, the robot trajectories are chain of coordinates to which they are moved. The first step, for all k robots sequentially compares all points of the paths and when coordinates match in time, the collision avoidance algorithm is called.

To adapt the collision avoidance algorithm, the ρ coefficient is used. It is applied to compare with the number of allowed collisions for each robot ($CS_k \leq \rho$). Robot at the time of motion has no more than ρ resolved collisions with an unknown number of collisions on the path as a whole. When the robot pass the resolved collision, it calculated next one and correct trajectory if required. If map changing with high frequency this approach allow to reduce the calculating of collision resolution. The coefficient ρ shows a conditionally static section of the path. The flowchart of the adaptation algorithm is shown in Fig. 1.

The collision avoidance algorithm was developed by the authors and described in [11]. The collision avoidance algorithm finds the collision area of two or more robots. Further, it find the free space, through which a new trajectory can be laid with minimal changes to the planned path. In this paper,

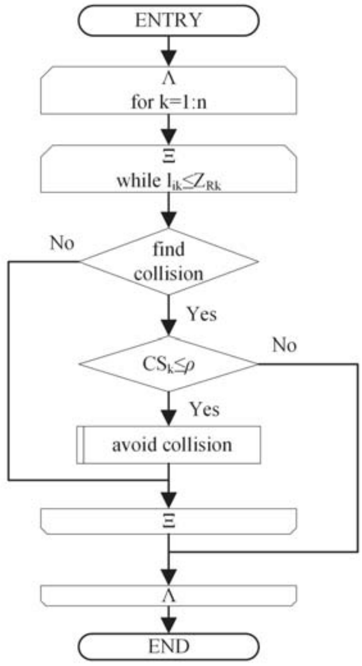


Fig. 1. Adaptive collision avoidance algorithm

a method of adaptation based on the environmental conditions is designed.

The algorithm of trajectory planning based on breadth-first search and is not a subject of research in this paper.

V. Modeling

A. Conditions of the simulation

To develop the collision avoidance adaptive algorithm we studied actual range of collision implementation. Table I shows the simulation parameters.

TABLE I. Simulation parameters

Parameter/Constant	Value	Units of measurement
Map size	50×50	u^2
the Number of robots	20	u
Visibility radius	80	u
Communication range	80	u
Obstacles on the map	25	%
Rate of map changes	0; 0.04; 0.4; 1; 2; 3; 4; 5; 10; 20; 30; 40; 50; 60	% per step
the Number of simulation cycles	150	u
Simulations	50	-

The aim of the first stage is the experimental determination the minimum range of the individual robot collision avoidance – ρ. The modeling environment has been launched to count and record collisions occurred with each robot and the consequent calculation of collisions recurring from step to step is carried out.

The aim of the second stage is to develop and research the collision avoidance adaptive algorithm by using the coefficients obtained in the first stage.

The main goal is to minimize the number of calls of collision avoidance function:

$$\forall R_k \in R[\lim{}_{time \rightarrow end} \sum (CS_i) \leftarrow \min \&(CS_i) \leq \rho],$$

where CS – call of collision avoidance function, ρ — coefficient of matching frequency of map changes and actual range of collision implementation.

B. Model

To examine the effectiveness of the adaptive algorithm the data obtained in simulation without the algorithm and with the range of collision implementation have been compared.

The initial position of the robots and goals on the map and replacing the obstacles are specified at random. Every robot constantly communicate with the others and fully update the map at each step of movement. Fig. 1 shows the general view of the interface of the modeling environment.

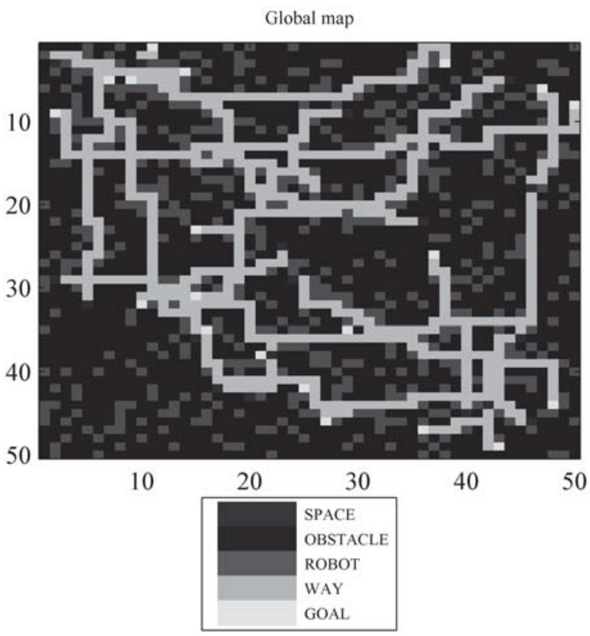


Fig. 2. The interface of the modeling environment

As was mentioned earlier, the map is a region marked up by a regular grid. Each cell can be either the impassable obstacle, or the passable area. In the simulation, the obstacles move / disappear / appear depending on the specified parameters from Table I.

The developed model consists of the following modules:

1) map display module (module shows obstacle, trajectories, robots and target points. The example in Fug. 1);
2) dynamic environment change module (module changes position of the obstacles on the map);
3) environmental analysis module (search the difference between maps on current and previous step);
4) robot motion control module (simulate robot movement from step to step);
5) trajectory algorithm (planning path from current position to the target);
6) collision avoidance algorithm / adapted collision avoidance algorithm;
7) module for planning group operations (distributes targets between robots).

The interconnection of the modules is shown in Fig. 3.

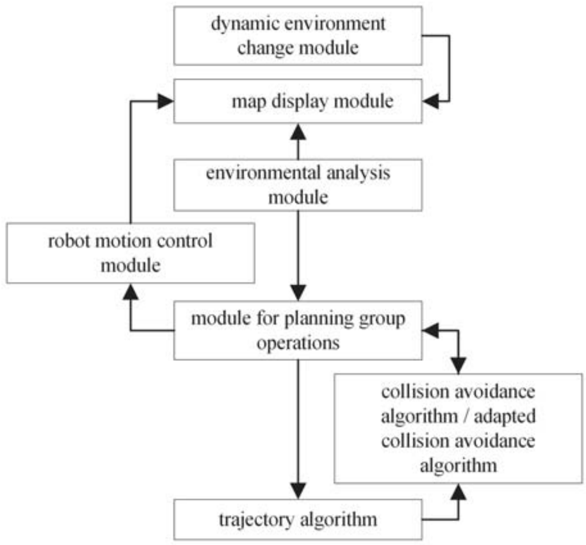


Fig. 3. Interconnection of the module

The simulation environment was designed in MatLab with build-in functions *spy* and *imagesc* being used to display graphics.

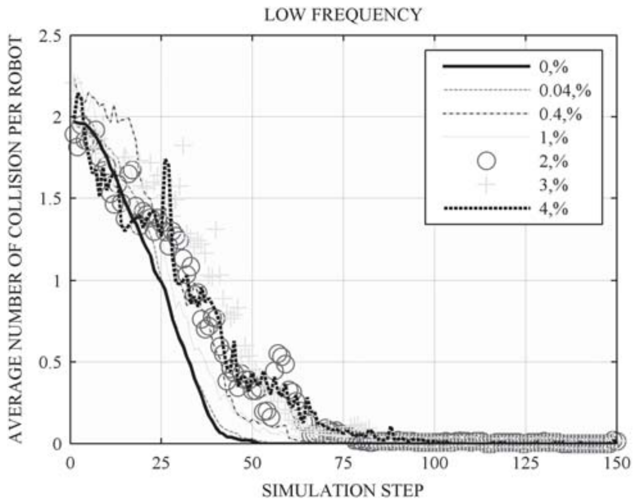
VI. Simulations

A. Collision research

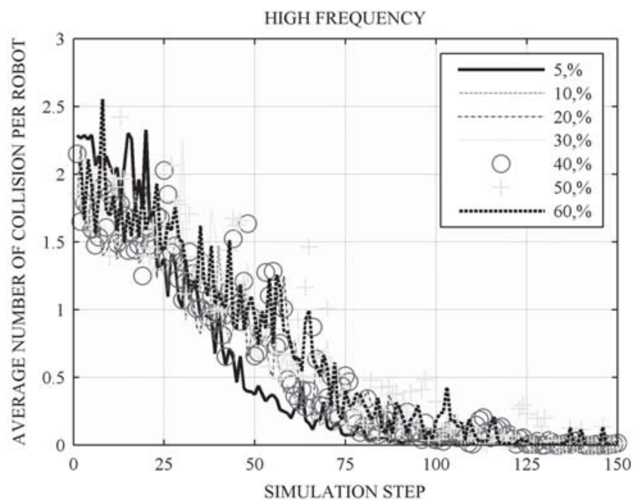
Two stage simulation was conducted. The first step determines the range of collision avoidance – ρ, for each robot. In other words, conditional static states of a dynamic environment were determined for each frequency change of the map. Fig. 4 and 5 are obtained based on the parameters in table I by simulating environment, counting collisions and calculating the number of recurring collisions from step to step for each robot.

Determination of the coefficient ρ will limit the number of calls to the collision avoidance function depending on the environment dynamics. ρ implements the analogue of lazy evaluation. Avoiding collisions the trajectories of the robots occurs only on the autonomous movement of the robot. The length of the trajectory of the autonomous movement depends on the frequency of the environment changes.

Fig. 4 shows curves of the maximum number of collisions with respect to the frequency of changes in the environment.

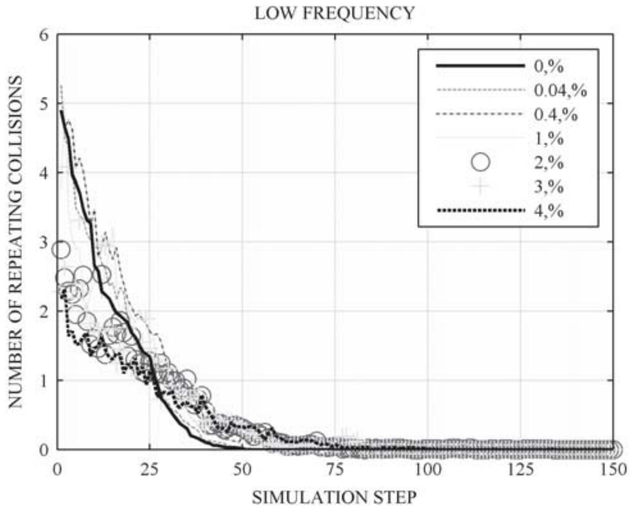


(a)

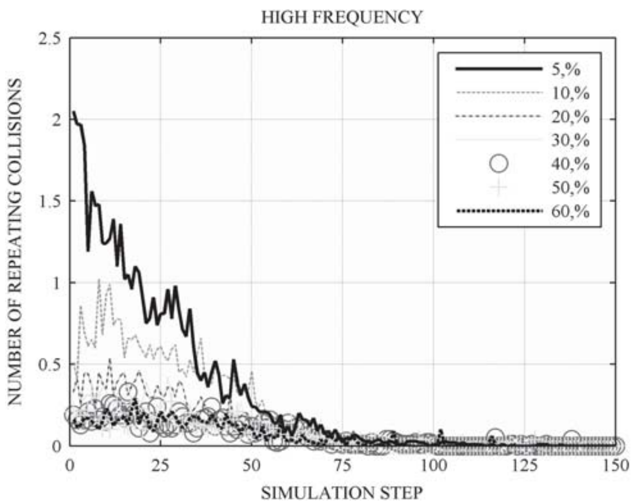


(b)

Fig. 4. Average number of collision per robot with respect to simulation step; a – low frequency; b – high frequency



(a)



(b)

Fig. 5. Number of repeating collisions with respect to simula-tion step; a – l o w frequency; b – high frequency

Fig. 4a environment conditions results in fluctuations at the graph. Fluctuation frequency and their magnitude increase with increase in frequency environment change. The most significant fluctuation difference can be seen at $\omega = 0$ and $\omega = 60$ marked by bold lines in Fig. 4a and 4b.

Fig. 5 shows graphs of recurring from step to step collisions, 5a shows the results at a low map dynamic (0 - 4% per simulation step), 5b – at a high frequency (5 – 60% per simulation step). The frequency of 5% can be considered as a limit value: at higher frequencies less than one collision recurs for a given model of the environment and the number of robots.

The graph at $\omega = 0$ can be considered as the upper limit of assessment: in this case the number of repeating from step to step collisions will decrease as the robots pass through the areas with collisions. At maximum frequency change will be minimal, but the number of collisions at each step will remain the same.

Fig. 5 shows as the frequency of map change increases, the frequency of collisions decreases, even though the number of collisions decreases according to Fig. 3 more slowly than with the static map.

Based on Fig. 4 and 5 and data averaging, we obtain table II, in which a certain frequency matches at the upper limit, the number of collisions necessary for calculating to minimize the amount of computation.

TABLE II. Collision avoidance range

Frequency, ω	0	0.04 - 0.4	1-2	3-5	10 - 60
Range, ρ	5	4	3	2	1

To determine the actual range of collision implementation, let us consider the intersections of the repeating collision curves at frequency $\omega = 0$ and the remaining frequencies.

B. Develop and research of the adaptive collision avoidance algorithm

Dependence of the collision avoidance range on the frequency of the environment obtained in section "Collision research" is used as a limit value when deciding on the end of collision calculation. Collisions, with low probability, are not calculating, which saves computing resources.

Adaptation occurs according to the following principle: for each robot, the number of allowed collisions does not exceed the value specified in the "Range" column of Table II. The flowchart of the developed algorithm is shown in Fig. 1.

The data on the range of the collision avoidance in table II are used in the comparison block $CS_k \leq \rho$. Also, the values of the resolution range of the collisions $\rho = \{1, 2, 3\}$ fixed for all frequencies were researched.

For simulation, the parameters from table I were used. Robots move in the dynamic environment and rebuild their paths if the changes on the map have influenced their own trajectory.

Fig. 6 shows set of curves with different parameters of the frequency of environment change with respect to the total number of calls to the collision avoidance algorithm with or without adaptation and the fixed range.

It is important to note that the curves in Fig. 6 differ from each other (in terms of the structure of the model and modules) only by the coefficient ρ. In case of "without algorithm", we can conditionally assume that ρ tents to infinity.

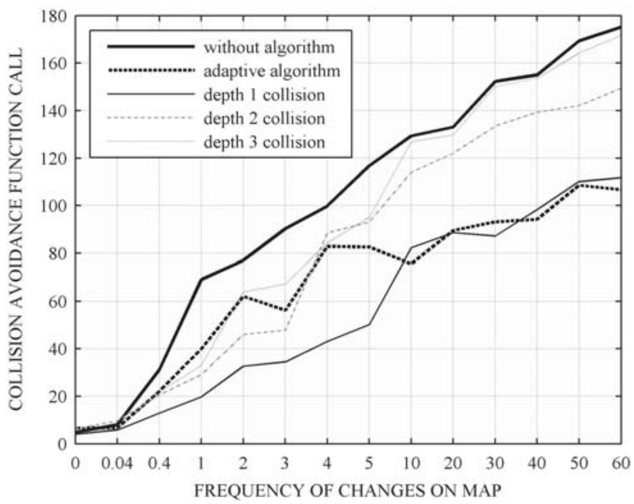


Fig. 6. Collision avoidance function calls

Let's assume the number of collision resolution calls without adaptation (solid black line in Fig. 6) has maximum computational complexity. If 25% of the map area is filled with obstacles and frequencies range from 0 to 60%, the number of calls is expected to increase in case frequency increases. Moreover, even at low frequencies, the number of calls increases significantly. For example, if there is 5% of the changing obstacles on the map, the number of calls exceeds 100, which is twentyfold of that on a static map.

Three graphs in Fig. 6 show the dependences of the map dynamics for cases of using the algorithm with adaptation for a fixed number of range: 1 – black solid, 2 – light gray dot-dashed, 3 – dark grey dotted.

Using the fixed range of the adaptive algorithm for calculating for one collision per robot, the computation volume was halved (1.97 times), with the factor equaling 2.25 at a low frequency, and 1.68 at a high frequency. For a fixed range of two and three collisions, the calculations decreased by 1.30 times and 1.21 times correspondingly.

Approximation of the plot of the three collision calculation to the graph of the full calculation of all collisions can be is explained by the fact that the average number of collisions per robot is slightly higher than 2 with the parameters of the environment and the group being set.

The bold dashed curve in Fig. 6 represents the plot of the adaptive algorithm with the range coefficients from table II. The course of the curve is similar to the one's for a fixed number of ranges of their application. The use of the adaptive algorithm made it possible to reduce the amount of calculations throughout the entire frequency range by an average of 1.44 times.

Comparing graphs with and without the use of an adaptive algorithm we can observe a significant reduction in the required computing power at high frequency map changes.

Fig. 4a shows a monotonically decreasing graphs, it means high implementation of the computed collisions. It indicates the effectiveness of calculation the collisions to the range of the conditional static map and the motion trajectories. The small number of computed collisions provides a frequent calling to the collision avoidance algorithm and collision search function. In other words, adaptation based on the coefficients of table 6 allows the robot to increase move time in an autonomous mode without communication with the control center or other robots at low map dynamic and reduce the number of collision calculation with high spatio-situational uncertainty to values obtained at low environment change frequency in the calculation of all collisions occurring on a planned path.

VII. CONCLUSION

In this paper the research of the effectiveness of using adaptive collision avoidance algorithm in spatio-situational uncertainty has been conducted.

The study of determining the range calculation of collisions resulted in setting up a correspondence between the frequency of the map change and the number of collisions resolved for each robot in the course of motion planning presented in Table II. This is the range of conditionally static states of the map and provides the possibility of autonomous robot movement. This information is used for developing the adaptive algorithm.

Testing the developed algorithm was carried out by comparing the number of calculated collisions for systems: without using the adaptive algorithm; with the regulator adapted to the frequencies; and fixed range of resolved collisions for the whole series of frequencies.

The main result of the simulation is that the number of calculated collisions for the group of robots for the adaptive algorithm was reduced by one and a half times, which reduces the requirements for the computing resources of the hardware part of the control system of the robot group.

Acknowledgment

The research is supported by the government contract of Russian Federation #2.9198.2017/8.9.

References

[1] R. Alami, S. Fleury, M. Herrb, F. Ingrand, and F. Robert, "Multi-robot cooperation in the martha project," *IEEE Robotics Automation Magazine*, vol. 5, no. 1, pp. 36–47, Mar 1998.

[2] P. Svestka and M. H. Overmars, "Coordinated path planning for multiple robots," *Robotics and Autonomous Systems, Elsevier*, pp. 125–152, Mar 1998.

[3] A. Fujimori, M. Teramoto, P. N. Nikiforuk, and M. M. Gupta, "Co-operative collision avoidance between multiple mobile robots." *J. Rob. Syst.*, vol. 17, no. 7, pp. 347–363, 2000.

[4] J. Jin, Y. G. Kim, S. G. Wee, and N. Gans, "Decentralized cooperative mean approach to collision avoidance for nonholonomic mobile robots," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, May 2015, pp. 35–41.

[5] L. Wang, A. D. Ames, and M. Egerstedt, "Multi-objective compositions for collision-free connectivity maintenance in teams of mobile robots," in *2016 IEEE 55th Conference on Decision and Control (CDC)*, Dec 2016, pp. 2659–2664.

[6] L. Juntao, D. Tingting, L. Yuanyuan, and H. Yan, "Study on robot path collision avoidance planning based on the improved ant colony algorithm," in *2016 8th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)*, vol. 02, Aug 2016, pp. 540–544.

[7] X. Wang, M. Kloetzer, C. Mahulea, and M. Silva, "Collision avoidance of mobile robots by using initial time delays," in *2015 54th IEEE Conference on Decision and Control (CDC)*, Dec 2015, pp. 324–329.

[8] U. Borrmann, L. Wang, A. D. Ames, and M. Egerstedt, "Control barrier certificates for safe swarm behavior," *IFAC-PapersOnLine*, vol. 48, no. 27, pp. 68 – 73, 2015, analysis and Design of Hybrid Systems ADHS. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S240589631502412X

[9] W. Luo, N. Chakraborty, and K. Sycara, "Distributed dynamic priority assignment and motion planning for multiple mobile robots with kinodynamic constraints," in *2016 American Control Conference (ACC)*, July 2016, pp. 148–154.

[10] F. Cherni, Y. Boutereaa, C. Rekik, and N. Derbel, "Autonomous mobile robot navigation algorithm for planning collision-free path designed in dynamic environments," in *2015 9th Jordanian International Electrical and Electronics Engineering Conference (JIEEEC)*, Oct 2015, pp. 1–6.

[11] P. S. Motorin D.Ye. and K. L.M., "An algorithm for collision avoidance in path planning for a group of robots in a spatio-situational indeterminacy," *St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunications and Control Systems*, vol. 10, no. 2, pp. 32–44, 2017.