Proceedings of the RAAD 2013
22nd International Workshop on Robotics in Alpe-Adria-Danube Region
September 11-13, 2012, Portorož, Slovenia

Minimum Startup Delay Approach to Coordination of Multiple Mobile Robots Motion Along Predefined Paths

Toni Petrinić[1], Mišel Brezak[2] and Ivan Petrović[2]

[1]*HEP-ODS d.o.o. Elektroprimorje Rijeka, Pogon Cres-Lošinj, Dražica 10, 51550 Mali Lošinj, Croatia*
E-mail: toni.petrinic@gmail.com
[2]*Department of Control and Computer Engineering, Faculty of Electrical Engineering and Computing, Unska 3, 10000 Zagreb, Croatia*
E-mail: {misel.brezak, ivan.petrovic}@fer.hr

Abstract. This paper is concerned with the collision free motion coordination of multiple mobile robots sharing the same workspace. To successfully coordinate motion of multiple mobile robots, a common approach is to assign priority level to each mobile robot. A mobile robot with highest priority takes into account only static obstacles while other mobile robots have to take into account both static obstacles and dynamic obstacles, which are the mobile robots with higher priorities. Avoiding dynamic obstacles is based on avoiding time-obstacles in the collision map. A mobile robot efficiently avoids the time-obstacles by minimum startup delay time of its original trajectory along the predefined path. By applying the same principle to all lower priority robots a collision free motion coordination of multiple mobile robots can be achieved. A proposed method is fast and more intuitive then delaying a nonlinear traveling length versus sampling time curve or mixed integer linear programming formulation, which checks overlap between two intervals of any collision-time interval pair.

Keywords. Multiple mobile robots coordination, Decoupled method, Fixed path coordination, Time-obstacles

1. Introduction

Many practical applications of autonomous mobile robots require the use of multiple robots. The range of applications includes toxic residues cleaning, transportation and manipulating of large objects, alertness and exploration, searching and rescue tasks and simulation of biological entities behaviors. The usage of multiple robots has several advantages compared to a single robot. For instance, the tasks that can be accomplished are inherently more complex than those that can be accomplished by a single robot. Also, the system becomes more flexible (since robots can have a variety of roles, the same group of robots can be employed for many different objectives) and robustness (system can be designed so that a robot can take over the tasks of another robot in case of failure).

Since the 1980s, researchers have addressed many issues in multiple robots, such as control architectures, communication, task allocation, swarm robots, learning, and so forth (Parker, 2008). A very critical issue is coordinating the motions of multiple robots interacting in the same workspace (Hwang and Ahuja, 1992; LaValle, 2006). Regardless of the mission of the robots, they must be able to effectively share the same workspace to prevent interference between the robots.

Coordination of multiple robots can be categorized in various ways: (i) completeness - whether they are guaranteed to find a solution if one exists, (ii) complexity - the computational requirements of the search process, and (iii) optimality - the quality of the resulting solution. Often, techniques that are complete and optimal are computationally too intensive to be used in practice. Alternatively, techniques that achieve computational tractability typically trade off optimality and/or completeness.

Two classes of methods (Todt et al., 2000) have been proposed for coordination between multiple robots in the same workspace. Centralized methods compute the paths for all robots simultaneously. These methods can find optimal solutions at the cost of being computationally demanding, usually making them unsuitable for satisfying the real-time applications. Decoupled methods compute the path for each robot independently ignoring the effects of any other robot and try to coordinate the resulting motions. These methods are often computationally much simpler than centralized methods but the resulting paths can be far from optimal and are not usually complete (Sanchez and Latombe, 2002).

An advanced solution of decoupled motion coordination is to design fixed geometry trajectories for all robots neglecting possible collisions with other robots, and then ensure collision avoidance by inserting a startup delay on original trajectories. This is not computationally demanding and allows that a lower priority robot passes safely through collision areas. Two advanced startup delay approaches are those based on a nonlinear traveling length versus sampling time (TLVST) curve proposed in (Kant and Zucker, 1986; Lee and Lee, 1987; Chang et al., 1994; Park and Lee, 2006) and on mixed integer linear programming (MILP) formulation proposed in (Akella and Hutchinson, 2002).

In this paper we propose a new approach to coordination of multiple mobile robots motion based on minimum startup delay time. It is faster and more intuitive than aforementioned approaches. Also we present an approach to safety passing through a long narrow corridor when the motion directions of the colliding robots are identical. Approach is based on minimal time-headway spacing policy between two colliding robots.

The rest of the paper is organized as follows. Chapter 2 presents an overview of decoupled methods. Chapter 3 brings problem formulation, construction of time-obstacle and collision map and criteria for minimum startup delay time. Simulation results are presented in chapter 4. Conclusions are drawn in chapter 5.

2. Decoupled motion coordination methods

A decoupled method for motion coordination of multiple robots in the same workspace reduces the time complexity for finding a solution, which makes it more practical in applications comparing to centralized methods. Instead of searching for a solution to the entire problem, it is divided into smaller problems that can be solved independently in less time. This comes at the cost of losing completeness and optimality. A decoupled method may not find a solution, even if it exists. In decoupled planning, two approaches exist (LaValle, 2006), namely prioritized planning and fixed path coordination.

Prioritized planning approach is originally proposed by Erdmann and Lozano-Perez, 1987. In this approach, priorities are assigned to each robot. A path is planned for the first robot using any single-robot path planning approach. The path for each successive robot A^i, then takes into account the plans for the previous robots $A^1,...,A^{i-1}$, treating these higher priority robots as dynamic obstacles. A potential field based algorithm for coordinating prioritized robots is presented by Warren, 1990.

Prioritized planning consists of two steps. Firstly, determination of an efficient priority order and, secondly planning a collision free trajectory in the presence of other (existing) trajectories. The choice of priorities significantly impacts on solution quality (van den Berg and Overmars, 2005). One can optimize the choice of priority by searching the space of all assignments (Bennewitz et al., 2002) but such a search is computationally expensive and does not guarantee finding a solution if one exists. The trajectory of a robot with higher priority can cause that a lower priority robot is not able to find a feasible trajectory. In such cases the trajectory of the higher priority robot needs to be revised, in order to find solutions for all robots (Zhu et al., 2013). This procedure requires more processing time. A dynamic priority strategy in decentralized motion planning for formation forming of multiple mobile robots is presented by Liu et al., 2009.

Fixed path coordination approach decouples the planning problem into path planning and velocity planning (Kant and Zucker, 1986). The path planning step first generates path for all robots independently, using a path planning method for single robots in environments of static obstacles (e.g. roadmap, cell decomposition, potential fields). The second step, velocity planning, plans a velocity profile that each robot should follow along its path so as to avoid collisions with other robots. Note that the paths planned in the first step are not altered in the second step.

Researchers mostly use TLVST representation of collision area (Kant and Zucker, 1986; Lee and Lee, 1987; Chang et al., 1994; Park and Lee, 2006; Johnson and Hauser, 2012). Because of nonlinear characteristics of TLVST curve, it is not easy and intuitive to find minimum startup delay time to coordinate multiple mobile robots, so researchers focus on only few robots and few collision areas. Johnson and Hauser, 2012, presented a computationally very demanding optimal, exact, polynomial-time planner for optimal bounded-acceleration trajectories along a fixed, given path with dynamic obstacles. However, it assumes straight path and used a simple double integrator to model robot dynamics. Akella and Hutchinson, 2002, developed an MILP formulation for the trajectory coordination of large numbers of robots by only changing robot start times (see Fig. 1).

Komlosi and Kiss, 2011, represent collision area of two robots with, so called, time-obstacle. The main advantage of that collision representation, which is used to find minimum startup delay time, is linear presentation of time-distribution of original trajectory along predefined path.

Note that fixed path coordination approach allows only for a robot to change its time-distribution along a predefined path. This method is very efficient with respect to processing time, while the resulting trajectories can be quite inefficient, since avoidance in time is used, while avoidance in space is ignored.

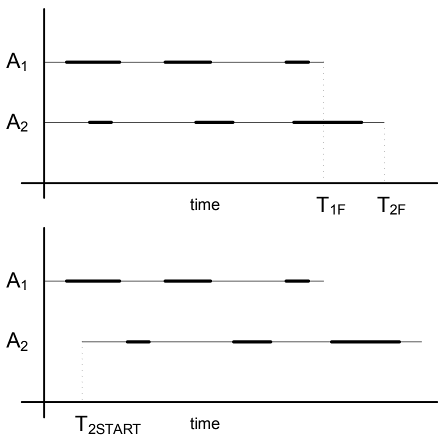

Fig. 1. Above: timelines for robots A_1 and A_2 with multiple collision intervals. Below: collision-free timelines for robots A_1 and A_2 being delayed at its start (example taken from Akella and Hutchinson, 2002).

3. Proposed motion coordination approach

The goal of the proposed motion coordination approach is to find minimum startup delay time of original time-distribution for each mobile robot (except for the robot with highest priority) so that no collisions between robots occur.

3.1. Problem formulation

Suppose that m mobile robots share the same planar workspace $W = R^2$. Each robot A^i, where $1 \le i \le m$, has its associated configuration space C^i, that is the set of all robot configurations. The robot configuration $q^i \in C^i$ contains complete specification of the pose of every point of the robot system - position $\left(x^i, y^i\right)$ and orientation θ^i.

By path we mean the geometric specification of a curve in configuration space:

$$\gamma : \zeta \in [0,1] \mapsto \gamma(\zeta) = q \in C. \tag{1}$$

A differentiable function τ given by:

$$\tau : t \in [0,T] \mapsto \tau(t) = \zeta \in [0,1], \tag{2}$$

with $\tau(0) = 0$ and $\tau(T) = 1$ is a reparameterization of the path γ. Time variable is t and T is a constant such that all robots complete their tasks in time T. A path together with the parameterization defines a trajectory. To simplify notation, we denote a trajectory as $\gamma(t)$. Robot velocity is specified a priori by specifying a time parameterization of trajectory:

$$v^i\left(t^i\right) = \frac{d}{dt^i}\gamma^i\left(t^i\right). \tag{3}$$

The optimal time-scaling algorithm (Brezak and Petrović, 2011a) is applied for trajectory planning along the predefined path in order to traverse the path in shortest time and to prevent wheel slip.

There are two sources of collisions in the workspace: robot A^i can collide with (i) static obstacle O, and (ii) robot A^j (index j denotes robot with higher priority level, $j < i$).

We make the following assumptions to generate a collision free coordination of the robot trajectories:

1. The only moving obstacles in the workspace are the robots.
2. Each robot does not collide with the other robots when they are at their start or goal configurations.
3. Each robot path is monotonic, i.e. the robot does not go backwards along its path.
4. The dynamics of each robot is known accurately.
5. Each robot executes its specified trajectory, once it starts moving.
6. The starting and final velocities of each of the robots are zero.
7. The robot motions are sampled at sufficient resolution so that no collisions occur during the motion between successive collisions free configurations.
8. Only one robot can be in collision area at a time.

Suppose that for robot A^i exists a path $\gamma^i\left(\zeta^i\right)$ such that no collisions between robot and static obstacles occur:

$$A^i\left(\gamma^i\left(\zeta^i\right)\right) \cap O = 0. \tag{4}$$

Now we consider collisions with dynamic obstacles, i.e. with mobile robots with higher priorities. A collision between i-th robot and all possible dynamic obstacles is given by:

$$A^i\left(\gamma^i\left(t^i\right)\right) \cap \left(\bigcup_{j=1,\dots,i-1} A^j\left(\gamma^j\left(t^j\right)\right)\right) \ne 0. \tag{5}$$

To avoid collision we change the time-distribution of original trajectory, i.e. find minimum startup delay $T^i_{\min}$:

$$A^i\left(\gamma^i\left(t^i \cdot e^{-T^i_{\min}}\right)\right) \cap \left(\bigcup_{j=1,\dots,i-1} A^j\left(\gamma^j\left(t^j\right)\right)\right) = 0. \tag{6}$$

Moreover, it is possible to find optimal priority order for all robots such that the total execution time for the entire group of robots is minimized. To do that it is necessary to change priority order each time and execute algorithm m! times, where m is a number of robots.

3.2. Construction of time-obstacles and collision map

A time-obstacle is always assigned to a collision area around the intersection of two geometric paths. Each collision area along the path of one robot generates a time-obstacle. The time-obstacle assigned to a collision area is the set of those $t^j \in \left[t^j_{enter}, t^j_{exit}\right]$ and $t^i \in \left[t^i_{enter}, t^i_{exit}\right]$ pairs (index j denotes robot with higher priority level, $j < i$) where any parts of two robots are located at the same place. We have to take into account the real physical dimensions of the robots, which mean that the entering time to a collision area is when the front of the robot enters the collision area, and the exit time is when its rear point exits it. It is assumed that each mobile robot has a physical safety area, which is represented by a 2-D circle of center (x, y) located in the point that represents robot's position and of radius R. The potential collision between i-th and j-th robots occurs when distance between two robots is less than or equal to the sum of their safety radius:

$$\sqrt{\left(x^i(t) - x^j(t)\right)^2 + \left(y^i(t) - y^j(t)\right)^2} \le R^i + R^j . \quad (7)$$

If the robot with lower priority cannot enter the area until the higher priority robot has passed through it, the time-obstacle will be rectangle shaped with vertexes $\left(t^j_{enter}, t^i_{enter}\right), \left(t^j_{enter}, t^i_{exit}\right), \left(t^j_{exit}, t^i_{exit}\right)$ and $\left(t^j_{exit}, t^i_{enter}\right)$ where t^i_{enter} and t^i_{exit} are the time values of entering and exiting the collision area for the lower priority robot, and t^j_{enter} and t^j_{exit} for higher priority robot. Fig. 2 shows colliding robots in the workspace and the time-obstacle of the colliding robots.

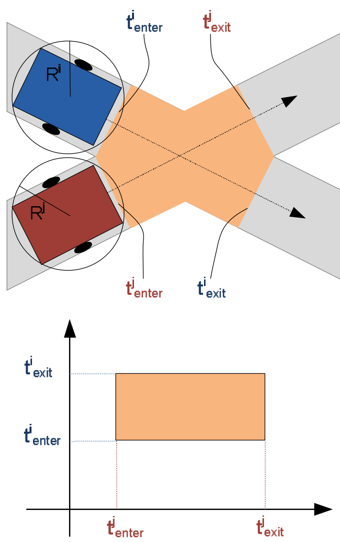

Fig. 2. Colliding robots in the workspace (above) and the time-obstacle of the colliding robots (below).

A collision map (see Fig. 3) contains one or more time-obstacles (red rectangles in Fig. 3). On Y-axis of collision map are lengths that represent time passage of i-th robot through n-th collision area. On X-axis of collision map are times that represent time passage of higher priority robots than i-th through the same n-th collision areas. Blue/green line with unit gradient in Fig. 3 represents the time scaling of original/delayed trajectory along predefined path (Komlosi and Kiss, 2011).

3.3. Criteria for minimum startup delay time

Avoiding time-obstacles includes collision detection of the time-distribution of original trajectory with time-obstacles. Finding minimum startup delay time is very simple in case of rectangular shape of the time-obstacles and linear unit gradient presentation of the original time-distribution. Each time-obstacle is translated on X-axis of collision map. If the time-obstacle is rectangle shaped with vertexes $\left(t^j_{enter}, t^i_{enter}\right), \left(t^j_{enter}, t^i_{exit}\right), \left(t^j_{exit}, t^i_{exit}\right)$ and $\left(t^j_{exit}, t^i_{enter}\right)$, then its representation on X-axis is the line with the start point $\left(t^j_{enter} - t^i_{exit}\right)$ and the end point $\left(t^j_{exit} - t^i_{enter}\right)$. Minimum startup delay time for i-th robot is then given by difference set:

$$T^i_{\min} = \min\left(\begin{array}{l} [0,\infty) \setminus \\ \bigcup\limits_{j=1,\dots,i-1;\,k=1,\dots,n} \overline{\left(t^j_{enter,k} - t^i_{exit,k}\right)\left(t^j_{exit,k} - t^i_{enter,k}\right)} \end{array} \right), \quad (8)$$

where n denotes number of time obstacles between i-th and j-th robots. In Fig. 3 union of translated time-obstacles on X-axis is represented by orange lines. Green length represents a minimally delayed original time-distribution.

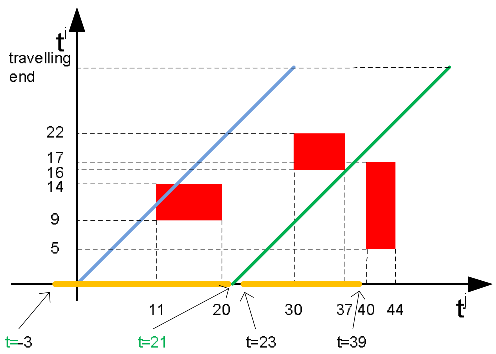

Fig. 3. Collision map with time-obstacles (red rectangles), time-distribution of original trajectory (blue line with unit gradient) and minimum delayed original time-distribution (green line with unit gradient).

When path geometries have common sections, e.g. long narrow corridor, and when the motion directions of the colliding robots are identical, it is senseless that the lower priority robot waits until higher priority robot passes through the long narrow corridor. In such a case, nonlinear time-obstacle characteristics $t^i\left(t^j\right)$ (blue curve in Fig 4.) can be approximated by a red parallelogram. Then minimal time-headway spacing policy between two robots through a common collision area can be applied. Let

t_H be a minimal time-headway. Green parallelogram with vertices $(x_1, y_1), (x_2, y_2), (x_3, y_3)$ and (x_4, y_4) is obtained by expanding red parallelogram with time-headway t_H. In Fig. 4, its translation to X-axis of collision map is represented with orange line with the start point $\min\left(t_1^j - t_1^i, t_4^j - t_4^i\right)$, and the end point $\max\left(t_2^j - t_2^i, t_3^j - t_3^i\right)$.

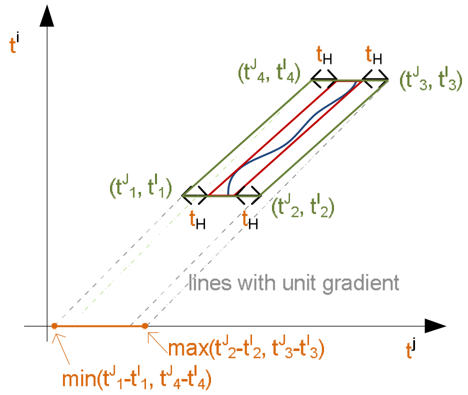

Fig. 4. Parallelogram shaped time-obstacle expanded with the minimal time-headway spacing policy between colliding robots.

In our analysis, we assume that each mobile robot executes its specified trajectory. However, it may not always be fulfilled so that a robot may have to stop due to emergency reasons. If the *i*-th robot stops in non-collision area, then lower priority robots freely proceed its travelling. If the *i*-th robot stops in collision area, then lower priority robots must wait until *i*-th robot continues its motion, or alternatively, trajectories can be replanned for other robots by considering the *i*-th robot as a static obstacle.

4. Simulation results

The simulation was performed in MATLAB®. The workspace and the paths of mobile robots system with four robots, used in simulation, are shown in Fig. 5. Robot path consists of lines and clothoids (Brezak and Petrović, 2011b). The path of a robot is defined by the initial robot pose and curvature profile (see Tab. 1). Curvature profile consists of segments with length $s_i[m]$ and end curvature $k_i(s_i)\left[m^{-1}\right]$. Initial curvature of the first segment is zero. The curvature changes linearly between two segments.

A differential drive robot that has two driving wheels and two castor (passive) wheels is used in simulation. The length and width of the robots are approximately the same and they both are 0.068 [m]. A physical safety circle with radius $R = 0.06[m]$ is assumed for each mobile robot. A dynamic model of the lightweight mobile robot takes into account velocity and acceleration limits, as well as limited adhesion force between robot wheels and the ground to prevent wheel slip, where it is considered that weight distribution on robot wheels varies due to inertial effects. Robot parameters and dynamic model of the robot are given in Brezak and Petrović, 2011a.

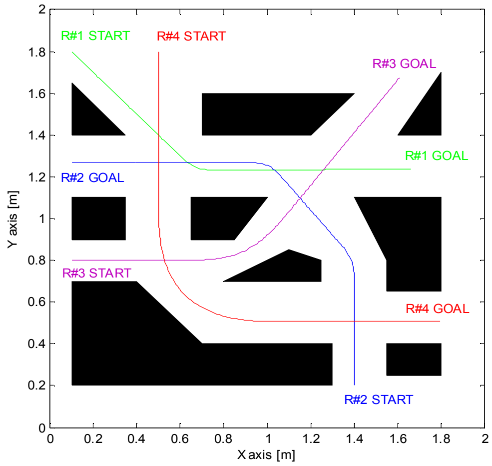

Fig. 5. Workspace and robots' paths used in simulation.

Tab. 1. Paths of the robots

Path / Robot	Initial robot pose: x[m], y[m], θ[rad]	Curvature profile parts: s_i [m], $k(s_i)$ [m^{-1}]
R#1	0.1, 1.8, -π/4	[0.7 0; 0.1 79; 0.1 -79; 0.9 0]
R#2	1.4, 0.2, π/2	[0.5 0; 0.1 70; 0.1 -70; 0.4 0; 0.1 87; 0.1 -87; 0.8 0]
R#3	0.1, 0.8, 0	[0.5 0; 0.3 10; 0.3 -10; 0.8 0]
R#4	0.5, 1.8, - π/2	[0.7 0; 0.5 6.3; 0.5 -6.3; 0.7 0]

The highest priority robot is R#1, the next one is R#2 then R#3 and R#4 has the lowest priority level. Time-optimal velocity along the path for robot R#1 is drawn in Fig. 6. Minimum startup delay time for R#2, R#3 and R#4 are shown in Fig. 7, Fig. 9 and Fig. 11, respectively. The original (blue line) and the minimum delayed (green line) time-optimal velocity profiles along the path for R#2, R#3 and R#4 are drawn in Fig. 8, Fig. 10 and Fig. 12, respectively.

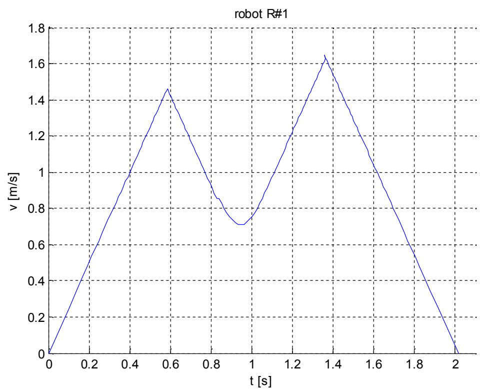

Fig. 6. Time-optimal velocity profile for robot R#1.

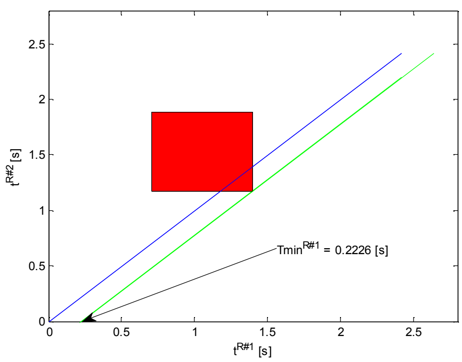

Fig. 7. Minimum startup delay time for robot R#2.

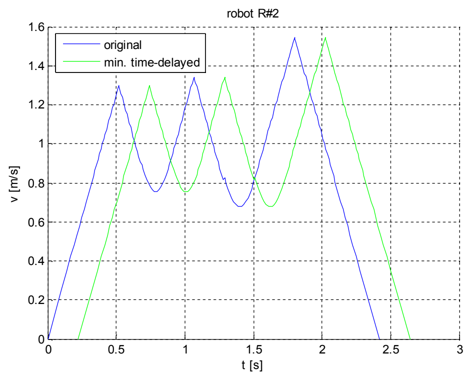

Fig. 8. Time-optimal (blue) and minimum startup delayed (green) velocity profile for robot R#2.

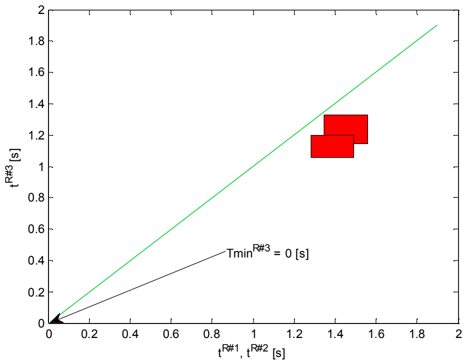

Fig. 9. Minimum startup delay time for robot R#3 (original and delayed time-distribution are overlapping).

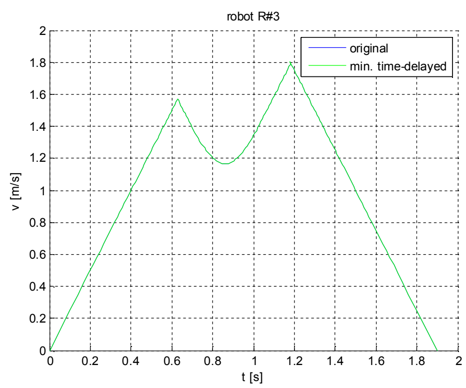

Fig. 10. Time-optimal (blue) and minimum startup delayed (green) velocity profile for robot R#3 are identical.

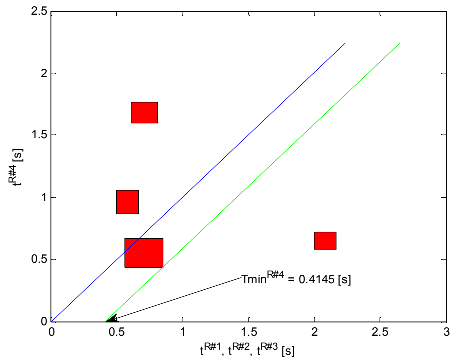

Fig. 11. Minimum startup delay time for robot R#4.

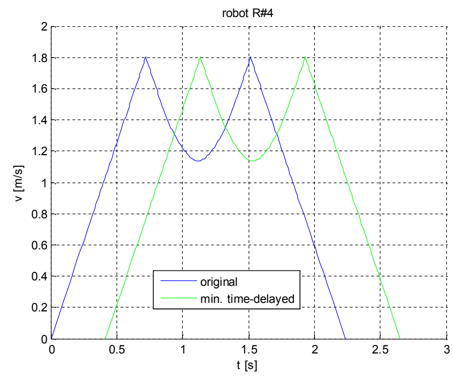

Fig. 12. Time-optimal (blue) and minimum delayed (green) velocity profile for robot R#4.

5. Conclusion

In this paper a new approach to coordination of multiple mobile robots motion along predefined paths is presented. The method is based on minimum startup delay time of original time-distribution. Important assumption for the successful motion coordination is that a lower priority robot waits until a higher priority robot passes through the collision area. The criterion for finding minimum startup delay time is proposed. A user may define a safety time-headway around any rectangle shaped time-obstacle if it is necessary.

In the future work we will analyze execution time of the proposed method and evaluate it through real-time experiments.

6. Acknowledgments

This research has been supported by the European Community's Seventh Framework Programme under grant No. 285939 (ACROSS).

7. References

Parker, L.E. 2008. *Chapter 40: Multiple mobile robot systems, Springer Handbook of Robotics*, Springer.

Hwang, Y.K. and N. Ahuja. 1992. Gross motion planning - a survey, *ACM Computing Surveys*, Vol. **24**(3), pp. 219-291.

LaValle, S.M. 2006. *Planning Algorithm,* Cambridge University Press.

Todt, E., G. Raush and R. Suarez. 2000. Analysis and Classification of Multiple Robot Coordination Methods. In: *Proc. of IEEE International Conference on Robotics and Automation (ICRA)*, Vol. **4**, pp. 3158-3163.

Sanchez, G. and J. C. Latombe. 2002. Using a PRM planner to compare centralized and decoupled planning for multi-robot systems. In: *Proc. of IEEE International Conference on Robotics and Automation (ICRA)*, Vol. **2**, pp. 2112-2119.

Erdmann, M. and T. Lozano-Perez. 1987. On Multiple Moving Objects, *Algorithmica*, Vol. **2**, pp. 471-521.

Warren, C.W. 1990. Multiple robot path coordination using artificial potential fields. In: *Proc. of IEEE International Conference on Robotics and Automation*, May, Vol. **1**, pp. 500-505.

Berg, J. van den and M. Overmars. 2005. Prioritized motion planning for multiple robots, *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Aug., pp. 430-435.

Bennewitz, M., W. Burgard and S. Thrun, 2002. Finding and optimizing solvable priority schemes for decoupling path planning for teams of mobile robots, *Robotics and Autonomous Systems*, Nov., Vol. **41**(2-3), pp. 89-99.

Zhu, Y., T. Zhang, J. Song and X. Li. 2013. A hybrid navigation strategy for multiple mobile robots, *Robotics and Computer-Integrated Manufacturing*, Aug., Vol. **29**(4), pp. 129-141.

Liu, S., D. Sun, C. Zhu and W. Shang. 2009. A dynamic priority strategy in decentralized motion planning for formation forming of multiple mobile robots, *IEEE/RSJ International Conference on Intelligent Robot and Systems (IROS)*, Oct., pp. 3774-3779.

Kant, K. and S.W. Zucker. 1986. Toward efficient trajectory planning: the path-velocity decomposition, *The International Journal of Robotics Research*, Vol. **5**, pp. 72-89.

Chang, C., M.J. Chung and B.H. Lee. 1994. Collision avoidance of two general robot manipulators by minimum delay time, *IEEE Transactions on Systems, Man and Cybernetics*, Mar., Vol. **24**(3), pp. 517-522.

Park, S. and B. Lee. 2006. A new analytical representation to robot path generation with collision avoidance through the use of the collision map, *International Journal of Control, Automation and Systems*, Feb., Vol. **4**(1), pp.77-86.

Lee, B.H. and C.S.G. Lee. 1987. Collision-free motion planning of two robots, *IEEE Transactions on Systems, Man and Cybernetics*, Jan./Feb., Vol. **17**(1), pp. 21-32.

Johnson, J. and K. Hauser. 2012. Optimal acceleration-bounded trajectory planning in dynamic environments along a specified path, *IEEE International Conference on Robotics and Automation*, May, pp. 2035-2041.

Komlosi, I. and B. Kiss. 2011. Motion planning for multiple mobile robots using time-scaling, *Mobile Robots – Control Architectures, Bio-Interfacing, Navigation, Multi Robot Motion Planning and Operator Training*, pp. 259-288.

Akella, S. and S. Hutchinson. 2002. Coordinating the motions of multiple robots with specified trajectories. In: *Proc. of IEEE International Conference on Robotics and Automation (ICRA)*, Vol. **1**, pp. 624-631.

Brezak, M. and I. Petrović. 2011a. Time-optimal trajectory planning along predefined path for mobile robots with velocity and acceleration constraints, *IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, July, pp. 942-947.

Brezak, M. and I. Petrović. 2011b. Path smoothing using clothoids for differential drive mobile robots. In: *Proc. of the 18th IFAC World Congress*, Milano (Italy), pp. 1133-1138.