

A Collision Control Strategy for Multiple Moving Robots

Pejman Kamkarian and Henry Hexmoor

Electrical and Computer Engineering Department, Southern Illinois University, Carbondale,
IL 62901, USA
Computer Science Department, Southern Illinois University, Carbondale, IL 62901, USA
pejman@siu.edu, hexmoor@cs.siu.edu

Abstract. We present a solution to prevent collisions among robots that are moving toward their respective goals. A robot may start moving at any time from its station to its goal. For a moving robot, the probability of conflict increases proportionately to the complexity of other robots' respective routes. In terms of lowering possibilities of collision, a proper strategy for controlling robot behaviors before encounters is essential. Prior research presented a negotiation-based solution through a broadcasting method. In our solution, we assume robots are unable or unwilling to negotiate or broadcast data among one another. They should possess a strategy to detect and predict conflict zones, and hence determine strategies to avoid collisions independently.

Keywords: Moving Robots, Robot Collision Control

1 Introduction

Imagine a factory floor with hundreds of unmanned guided vehicles rushing to complete tasks at disparate but route-crossing destinations. Similarly, on a busy street in a major metropolitan city like NYC, people mitigate collisions by the rapid uptake of others' trajectories (i.e., intention inferencing) as well as instantaneous velocity adjustments. People either mutually accommodate one another by modifying their own trajectory or select strategies from a predetermined set of conventions that minimize collisions. Negotiated trajectory adjustments for multirobots is one possible solution as found in [7]. The majority of negotiations in a multi-robot environment operate in multiphases where each robot shares a trajectory with all other robots operating in the common workspace. Motion planning methods provide mechanisms to detect potential collisions. In order to detect collisions, robots preplan the paths each robot will follow together with the associated trajectories along these paths. Straight line paths have been reported with advantages for multi-robot systems [1]. Another approach to multirobotic collision avoidance is found in [10]. Founded on A* algorithm, M* can be proven to find an optimal path in finite time. In this paper we offer an alternative solution. Section 2 provides our basic assumptions and section 3 defines a collision zone. Section 4 outlines a multiphase reasoning whereas section 5 describes validation on specific cases. Conclusions are drawn in section 6.

2 Concepts and Premises

In order to develop our strategy, we made following five main assumptions.

(1) As a typical mobile robot system, we assumed a set of four co-located robots, which are located at different four corners of a room. Each robot A_i has a separate destination assigned to it, which is located on the opposing perimeter of the room. Each robot independently operates on its task. There is no goal sharing among robots. Each destination cannot be assigned to more than one robot at a time. At each moment, the status for a robot is given by $A_i = \{P_i, V_i, \partial_i\}$, where P_i is the priority rate, V_i is the velocity at moment t , and ∂_i is the current direction of robot i . The priority rates vary for each destination and will be determined based on the local status of the tasks, goals, and robot circumstances. Initially, it can be a randomly assigned number to each robot. It can be as a rate based on the robot ability or task urgency for each robot. ∂_i is the destination i and direction to it.

(2) All robots are circular in shape and all robots to have same size [2], [5], [9]. For simplicity, robots can only move in a straight line forward. They are unable to move backwards or turn.

(3) Each robot will have a different trajectory to move towards its goal location. For example, robots will have a constant speed without stops before reaching their goals.

(4) Robots may or may not be able to directly communicate with one another. Broadcasting information is used by many multi robot systems such as [2] and [5]. Since our solution is a non-communication approach we are not focusing on the broadcasting data. Our robots are unable to share information.

(5) Each robot is equipped with a vision sensor, which enables them to detect other robots within a fixed distance. The sensor is located in front (i.e., the moving direction). Vision is used in many applications such as optical motion planning mapping [6], [8], or the motion sensors themselves [4], [3], [11].

3 Collision Zone

Since each agent is equipped with a vision sensor, it is able to detect objects in a certain range. Each agent is represented by a set of specifications as it denoted by $A_i = (S_i, R_i, D_i)_t$, where S_i represents speed of the i^{th} agent, R_i captures the maximum detection range, and D_i indicates direction of the relevant agent, at time t . In figure 1, we assumed two moving agents that encounter a collision zone.

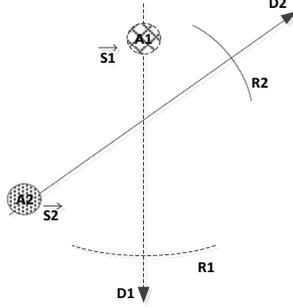


Fig. 1. Two moving agents in a collision zone.

By definition, a collision zone is formed when more than a single agent, which is moving toward its goal, detects and recognizes another moving agent. A collision incident will be considered by moving agents that detect collision zones and hence need to perform evaluation and decision for preventing collisions. Our strategy specifies a complete halt (i.e., full stop) for moving agents until the risk of collision expires. For simplicity, all moving agents have the same speed and hence take the same time to perform a full stop. However, in the real world, we will have different types of moving agents with different speeds. A faster speed means a greater distance to travel before a complete stop. Based on that, we considered a direct relationship between decision making when confronting a collision zone and velocity, reaction time, and stopping mechanism, as follows. $DM_i \propto (\alpha, \beta, \gamma)_i$. α represents robot speed, β is the reaction time, and γ denotes the stopping mechanism for the i^{th} moving agent. In order to adjust the reaction time we need to consider the processor in the agent, which leads us to focus on the mechanical properties. The velocity may vary based on the objective of the agent and the environmental specifications.

4 A Two Phase Solution of Collision Prevention

The main consideration of this paper is to develop a strategy that is able to control moving agents toward their goals with potential risks of collision. Each agent is in either one of three conditions at any time.

(i) Waiting status, which indicates the agent is either will not start to move from the start, or is already dwelling in a collision zone and is waiting for the agents with higher priority to vacate the zone,

(ii) Moving status represents that the agent is either free of collision zones or located in a collision zone but is permitted to continue moving toward its goal due to its priority that is higher than others in the zone,

(iii) Has reached its goal and hence removes itself from consideration.

The collision zone specifications, as well as conditions for entering to decision phase, are already discussed. In the remainder of this section, we propose a two phase solution to avert collision among moving agents while they are in a collision zone.

Decision making process consists of two separate processes of *detection* and *decision*. Each process is governed by its own rules and conditions and we will discuss them in more details next.

4.1 Detection

The aim of this phase is to determine collision zones for moving agents. As discussed earlier, each agent is equipped with a visual sensor. The agent detects moving agents when they are within detection range. The sensitivity of each sensor can vary based on the agent speed. The detection range must be long enough for a moving agent to stop in a reasonable amount of time before collision. Assume two agents $A_1, A_2 \in A_i$, detect one another while moving toward their goals with V_1 and V_2 indicating speeds respectively. If $V_1 > V_2$, which represents a higher speed for A_1 , the detection range R_1 , must increase enough to avoid collision.

4.2 Decision

A set of n robots enter a decision phase when they enter a collision zone. They have to decide which one is allowed to continue movement toward its goal while all others must stop and wait in the collision zone. Assume the set of n agents A_i , are present with a collision zone. Consider a flag with an identification value to represent the priority of that agent such that it is visible and hence can be detected using the vision sensor. The number is unique and reflects the robot urgencies for performing its task. At the beginning of the movement process priority is assigned and is determined by the following steps.

- (i) Priorities have to be comparable.
- (ii) They have to start based on their priorities.
- (iii) All values must be assigned to agents at the beginning of the general movements.
- (iv) Each priority remains the same until the task is completed.
- (v) The total number of agents are numbered from 1 to $|A| - 1$.

Each agent has a unique value and hence there is a set of n priorities at the beginning of movement process as follows: $P_i = \{P_1, P_2, \dots, P_n\}$, where P_n , is the priority of agent n^{th} agent. With a set of g agents in a collision zone, the priority set becomes $P_i = \{P_1, P_2, \dots, P_g\}$, where $(P_1, P_2, \dots, P_g) \in P_i$.

At each moment, an agent compares the set of priorities to recognize the agent with the highest priority. Once found, if the highest priority belongs to self, robot will continue toward its goal unabated; otherwise, it stops until the condition of collision changes by either a new agent entering to the current zone or the highest priority agent leaves the zone. The process then continues and allows the next highest priority agent to move toward its goal.

5 Solution Algorithm and Experimental Results

In this section we illustrate our solution strategy in detail. Figure 2 presents a general flowchart of collision control strategy.

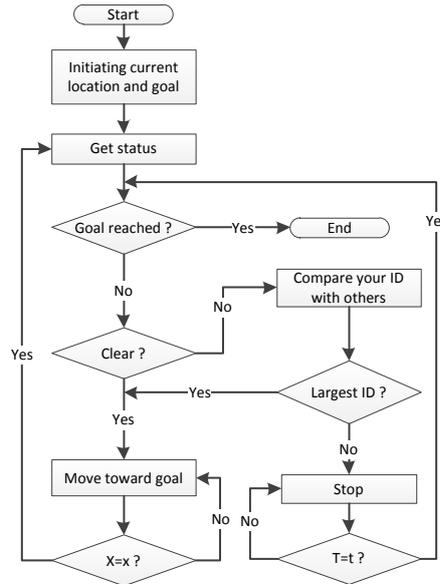


Fig. 2. General outline of collision control strategy in flowchart format.

Once each robot is assigned a goal it becomes activate and starts to move toward it. The process of movement continues until it reaches the goal. The mechanism of checking the status can be implemented by either distance or time measurements. In order to check status by distance, each active robot measures the status when passing a certain distance x . Active robots check the status when a certain time t has passed, if the checking status is based on the time. For simplicity, all robots may follow a single strategy for measuring status. However, in the real world they can follow different rules to measure status. Reasons are many folds. They can be made to perform different activities and goals. They can be made with different structures and speeds and hence need different times to reduce their velocities and to eventually stop. Each robot has a reference list of priorities for all other robots, at the beginning of movement process.

To implement our strategy we assuming in a two dimensional (2D) space where there four different robots parked. We also assume that each goal is located at the center of each edge. The goals will randomly be assigned to each robot and hence they start moving toward them consequently. Figure 3 shows the indicated situation for robots and their goals.

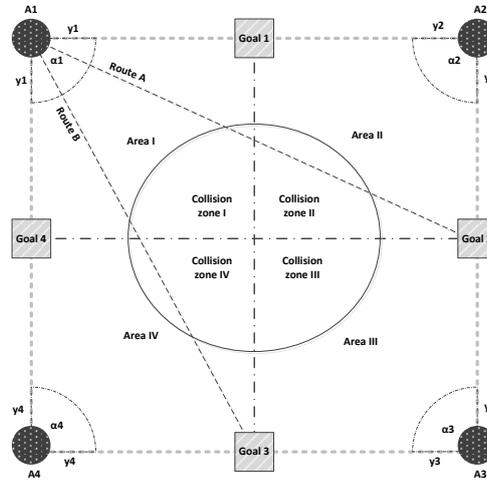


Fig. 3. 2D scope with 4 robots and 4 goals.

We considered robot movements only inside the 2D space and hence each robot can accept only the goals that are not located on the same edge with them and hence, they can accept between either one of 2 goals. For simplicity, we also assume that the robots only can move toward a straight line and are not able to change their route while moving toward their goals. Goal assignment can be without restrictions, in terms of sharing. On the other hand, it is possible to assign the same goal to more than one robot. Although in the real case, the proportion of numbers of robots and goals may be vary in our space we assume the number of robots is the same as the number of goals. The space that each robot needs to pass before reaching goals is the same. However, the required time before reaching the goals vary and will be determined by the number of collision zones that leads them to be stopped and wait until detection status indicated a clear status.

We considered the following algorithm as our solution. However, they can vary in different situations and robot constructions. The algorithm is given next.

Algorithm 1. Collision avoidance algorithm

1. Initialize current location.
2. Initialize goal location.
3. Set your movement direction toward goal.
4. Determine status from vision cameras.
5. If status is clear then go to step 7; otherwise, go to step 9.
6. If distance to the goal is negligibly small, stop.
7. Move toward goal.
8. If the traveled distance is x , then go to step 4; otherwise, go to step 7.
9. Initialize other robots inside the common collision zone.

10. Compare others' identifications with self.
 11. If self has the highest identification, then go to step 6; otherwise, go to step 12.
 12. Stop and count up wait time.
 13. If wait time reached t , then go to step 4.
-

The amount of needed time to perform a movement until reaching the desired goal will be larger when a larger number of robots with shared goals start to move toward their goals at the same time. To form our experimental analyses, we considered starting moving status, sharing goals conditions, as well as sharing areas probabilities. Figure 4 consists of three different series of experiments. The first series show the time diagram; when all robots start moving toward their goals at the same time, they do not share their goals with one another, and also none of them pass through other robot areas, during movement toward reaching its goal. In second series, however, robots might pass through areas belonging to one others with a 50% chance. In other words, at least one robot from any other area will pass through any other robot area. Series three, assumes each area will occupy by at least two other robots during experiment. On the other hand, the probability of passing other robots through each area is 100%.

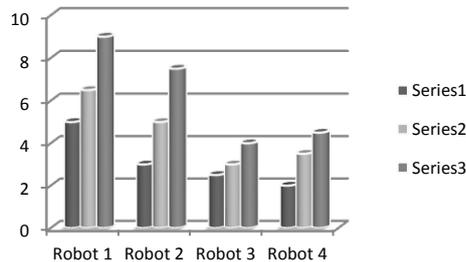


Fig. 4. Time diagram for robots when all robots start to move at the same time without sharing their goals and different situations for passing through one another's area.

For the second round of experiments, we assumed goals to be shared by more than a single robot. Figure 5 consists of three different series of experiments. The first series show the time diagram; when all robots start moving toward their goals at the same time, they share their goals to one other, however, none of them pass through other robot areas, during movement toward reaching its goal. In second series, however, robots might pass through areas belongs to one other with the 50% chance. In other words, at least one robot from any other area will pass through any other robot's area. Series three assumes each area will occupy by at least two other robots during experiment. On the other hand, the probability of passing other robots through each area is 100%.

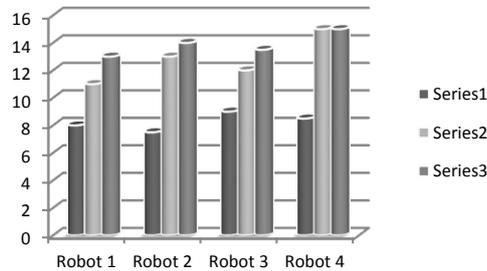


Fig. 5. Time diagram for robots, when all start to move at the same time, they occupy each goal with more than a single robot, and with different situations for passing through one another's area.

For the third round of our experiments, we assumed randomized scheduling time for the time that each robot starts moving toward its goal. Figure 6 consists of three different series of experiments. The first series show the time diagram; when all robots start moving toward their goals in a random time. They do not share their goals to one other and also none of them pass through other robot areas, during movement toward reaching its goal. In second series, however, robots might pass through areas belongs to one other with the 50% chance. In other words, at least one robot from any other area will pass through any other robot area. Series three, assumes each area will occupy by at least two other robots during experiment. On the other hand, the probability of passing other robots through each area is 100%.

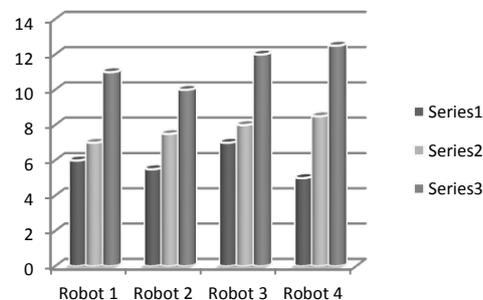


Fig. 6. Time diagram for robots when they start to move at random times without sharing their goals and with different situations for passing through one another's area.

For the final round of experiments, we considered the robots to start moving at a random time and occupy each goal by more than a single robot. Figure 7 consists of three different series of experiments. The first series show the time diagram; when all robots start moving toward their goals at a random time. The goals are shared between more than single robots. Also none of them pass through other robot areas, during movement toward reaching its goal. In second series, however, robots might pass

through areas belonging to one another with the 50% chance. In other words, at least one robot from any other area will pass through another robot's area. For series three, assume that each area is occupied by at least two other robots during the experiment. On the other hand, the probability of passing other robots through each area is 100%.

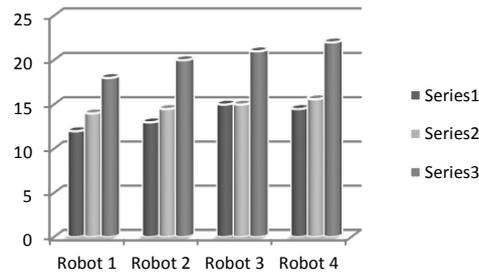


Fig. 7. Time diagram for robots, when they start moving at a random time, with sharing goals, and having different situations for passing through one other area.

6 Conclusion

This paper proposed a non-market based method to prevent multiple moving robots from collisions. Based on this strategy, the robots are able to decide how to avoid collision, without broadcasting data among them. This method is useful, when having different types of multiple moving robots that are not able to communicate with one another because of having some differences, such as different programmed or different type of usage. When a robot states that is in a conflict zone with many other robots, it compare its priority with all other robots in a same zone, and hence will decide, how to perform a decision based on avoiding collision.

References

1. J.D. Calliess, D. Lyons and U.D. Hanebeck, "Lazy Auctions for Multi-robot Collision Avoidance and Motion Control under Uncertainty", AAMAS Workshops 2011, pp. 295-312, ACM Press (2011).
2. C.M. Clark, S.M. Rock and J.C. Latombe, "Motion planning for mobile robots using dynamic networks", Proc IEEE Int Conf on Robotics and Automation (2003).
3. F. Expert, S. Viollet and F. Ruffier, "Outdoor field performances of insect-based visualmotion sensors", Journal of Field Robotics, vol. 28, no. 4, pp. 529–541 (2011).
4. J. Gaspar, N. Winters and J. Santos-Victor, "Vision-based Navigation and Environmental Representations with an Omni-directional Camera", IEEE Transactions on Robotics and Automation, Volume 16 Number 6, pp 890 -898 (2000).
5. Y. Guo and L. Parker, "A distributed and optimal motion planning approach for multiple mobile robots", Proc IEEE IntConf on Robotics and Automation, pp. 2612–2619 (2002).

6. S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning", *International Journal of Robotics Research*, Volume 30 Issue 7, pp. 846-894 (2011).
7. M. Krishna, S. Chellappa and H. Hexmoor, "Reactive Navigation of Multiple Moving Agents by Collaborative Resolution of Conflicts", *Journal of Robotic Systems*, Volume 22, Number 5, pages 249-269, Wiley Periodicals, Inc (2005).
8. F. Mazzini, D. Kettler, J. Guerrero and S. Dubowsky, "Tactile Robotic Mapping of Unknown Surfaces", With Application to Oil Wells, *IEEE Transactions on Instrumentation and Measurement*, vol. 60, pp. 420- 429 (2011).
9. P. Srivastava, S. Satish and P. Mitra, "A distributed fuzzy logic based n-body collision avoidance system", *Proc of the 4th Int Symposium on Intelligent Robotic Systems*, Bangalore, pp. 166–172 (1998).
10. G. Wagner and H. Choset, "M*: A Complete Multi robot Path Planning Algorithm with Performance Bounds", In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3260-3267, IEEE press (2011).
11. L. Zhang, T. Zhang, H. Wu, A. Borst and K. K"uhnlenz, "Visual Flight Control of a Quad rotor Using Bio inspired Motion Detector", *International Journal of Navigation and Observation*, Volume 2012, 9 pages, Hindawi pub (2012).