Social Reasoning and Collaboration among a Large Group of Robots

Henry Hexmoor and K. Madhava Krishna
CSCE Dept., University of Arkansas
Fayetteville AR 72701
{hexmoor, mkrishna}@uark.edu

Abstract

Collaborative technologies and reasoning strategies gain prominence with the growth in multi-agent systems, ubiquitous sensor systems and ubiquitous computing. This paper presents an architecture for social reasoning for agents who share the task of reconnaissance Collision conflicts amongst a large group of mobile robots is best resolved with social reasoning that involve negotiation and understanding.

1. Introduction

Unmanned vehicles (sometimes called robots) have become common solutions in the military as well as the industry. In the early days, robots were used as part of larger integrated solutions. Robots were given repeated tasks and predictable patterns of interaction. This may work in highly engineered domains as in manufacturing. However, in dynamic domains with large groups of robots, it is not practical to design for fixed interaction protocols. Instead, we must endow individual robots and the group as a whole, ability to reason about their own interaction pattern as well as alter the parameters of interactions in order to safe-guard the overall system performance. In this paper we discuss the foundations of such abilities and give results from our recent work on social reasoning about collision avoidance that improves overall group performance. For example, in cooperative collision avoidance (CCA) between robots often crisscross each other's path in rapid succession or come together to a common location in large numbers. CCA occurs in many coordination tasks such as among unmanned combat aircraft vehicles (UCAV). Cooperation between robots becomes mandatory when solutions for individual resolution of the conflicts are exhausted in the solution space. Robots then need to enter cooperation when individually they are unable to find a solution in the velocity space that could avoid the conflict (collision). A solution may not be found in the velocity space either because they do not exist or existing solutions lead to conflicts with other robots. We have developed a novel negotiation based scheme for multi-robot collision avoidance [1, 2] that involves cooperation between robots for decision-making as well as a provision for propagating conflicts or requesting robots not involved in a conflict to help resolving a collision conflict between two robots. We have also shown the utility for social functions such as *benevolence* in reducing the number of collision conflicts that arise in a multi-robot setting as well as depicting their role in decreasing the frequency of cooperation and negotiation amidst robots.

In the remainder of this paper we present architecture for collaboration and social reasoning. We describe two applications of this architecture in the areas of multi-robot collision avoidance and multi-robotic surveillance. Simulation results are presented to corroborate the efficacy of the architecture.

2. An Architecture for Social Reasoning and Collaboration

Outline of our architecture is shown in Figure 1. The architecture consists of three concurrent loops. These loops are independent and may not be synchronous. The social reasoning loop (topmost loop) serves to detect both the agent's relation with others in the agent's group as well as any other relationships among agents known to the agent. These relationships may be general ties or involve specific aspects of interaction such as trust between agents. Beyond relationships between individuals, an agent in our system constantly estimates its disposition toward its entire group. This loop is shown to feed into the middle loop. The middle loop is where an agent is primarily concerned with its course of action and moment-to-moment decisions as well as its utilities and goals. The top-level loop provides the status of its relationships with other agents. The bottom loop estimates the constancy in other agent's abilities, courses of action, as well as social relationships. Forecasting what other agents will do is important in decision-making.

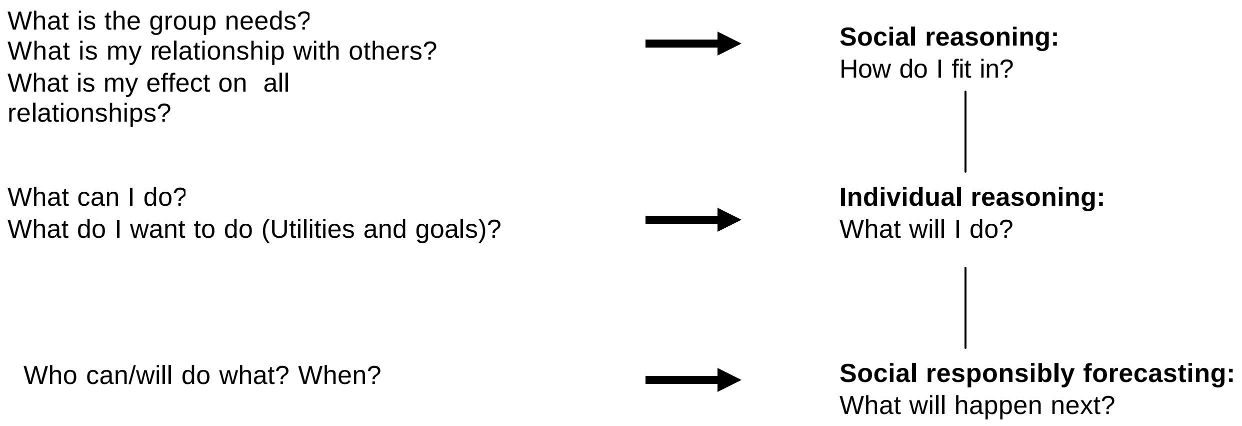


Figure 1. Architecture for multi-robot social group reasoning

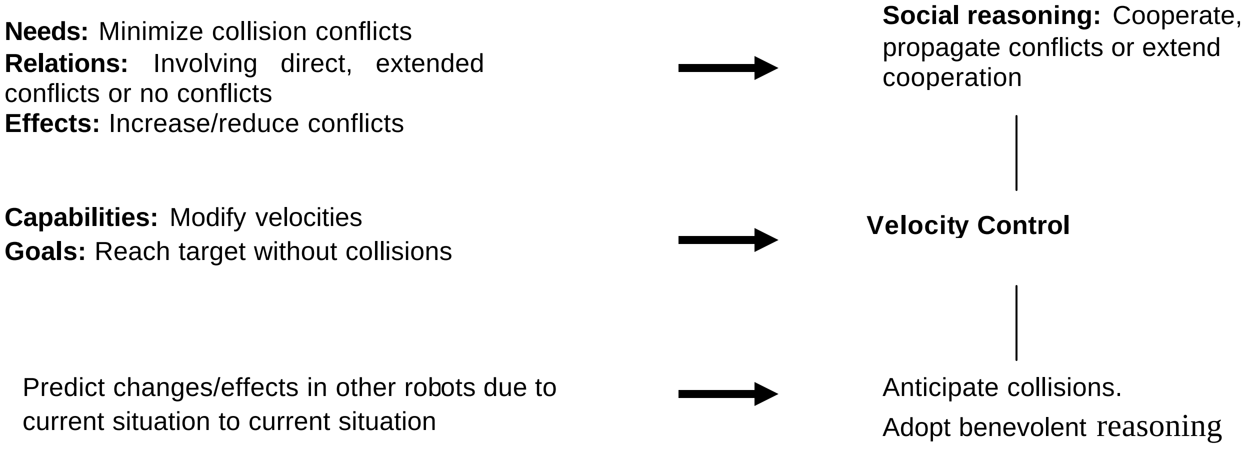


Figure 2. Adapting the architecture to multi robot collision avoidance system

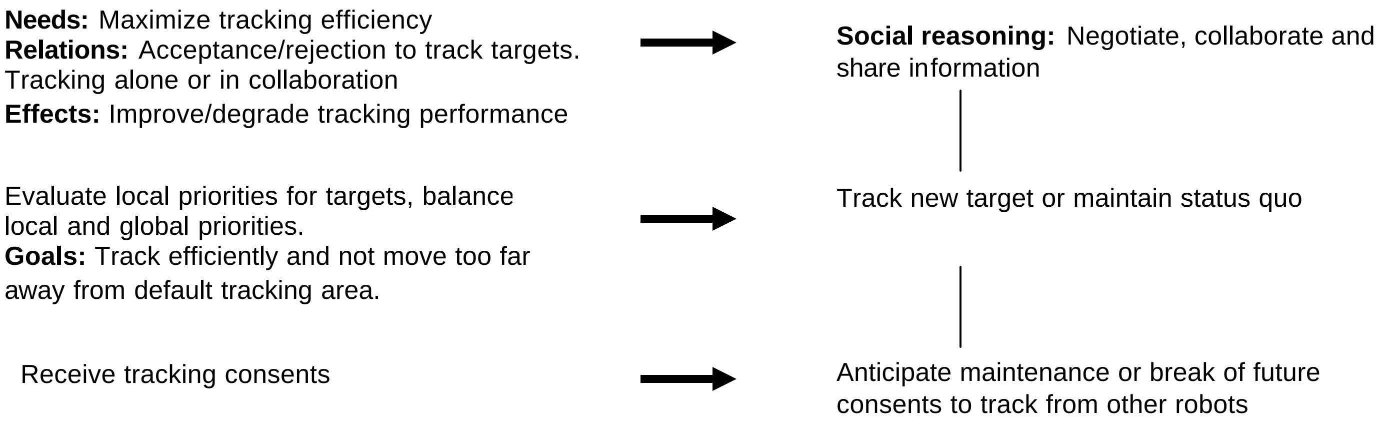


Figure 3. Adapting the architecture to multi robot surveillance and target tracking

3. Application of the Architecture to a Multi-Robot Collision Avoidance System

In this section we portray how the architecture of Figure 1 can be adapted to a system of several moving robots performing their tasks. While performing their tasks they come crisscross each other frequently and often in large numbers and CCA is inevitable in such situations. Figure 2 depicts the architecture modified to a CCA system.

The top most loop of the Figure concerns with placing the robot in the context of other robots and characterizes the state of the robot in terms of the relations it possesses with others. The relations could be one involving direct conflict or extended conflicts or no conflicts. These terminologies are elaborated in [2]. In brief a robot is in direct conflict at an instant if it has a collision conflict with one or more robots in the system at that instant. A robot is in extended conflict if it has no conflict with any of the robots involved in a direct conflict, however a modification of its velocity suitably could resolve conflicts between robots directly involved. The overall group need is characterized as minimizing the total number of conflicts. A robot also measures its own action as that of increasing or reducing conflicts. According to these criteria the robot reasons the need for cooperation, conflict propagation or to extend a lending hand although not involved in the conflict. Robots involved in a conflict directly cooperate to resolve it. If the resolution fails they propagate the conflicts to other robots not involved in the conflict directly that nevertheless could help in the resolution. A robot not involved in a conflict extends cooperation to resolve conflicts between robots involved in a direct conflict. At the middle level the robot concerns with its primary goal of reaching its target without collisions and makes use of its capability to modify its velocity to achieve this goal. The bottom loop of the architecture involves a robot reasoning about the changes or effects to the environment by predicting actions of other robots. A robot can anticipate collisions between two other robots that ***are not aware of the impending danger***. In such a case if the possibilities exist that the robot can act benevolently by modifying its velocity such that the collision gets averted without the other robots involved in the conflict ever getting aware of it.

3.1 Simulation Examples

The case of extended conflict and extended cooperation: Figure 4a shows an instant during the navigation of a system of five robots where robots 1 and 3 are unable to resolve their conflicts between them individually as well as cooperatively as cooperative solutions lead to indirect conflict with robot 4. Hence 1 and 3 propagate a request to resolve their conflict to 4 thereby embarking on the conflict propagation phase as the last attempt to resolve their conflicts. Robot 4 accepts requests from 1 and 3 and is able to solve the request of 1 by modifying its current velocity such that 1 manages to avoid its indirect conflict with 4. This scenario is depicted in Figure 4b where 4 moves faster in such a way 1 and 3 are able to avoid their mutual direct conflict.

Number of robots	Number of attempts at cooperation	Number of conflict propagations
10	2	2
15	4	3
20	8	4
30	12	5

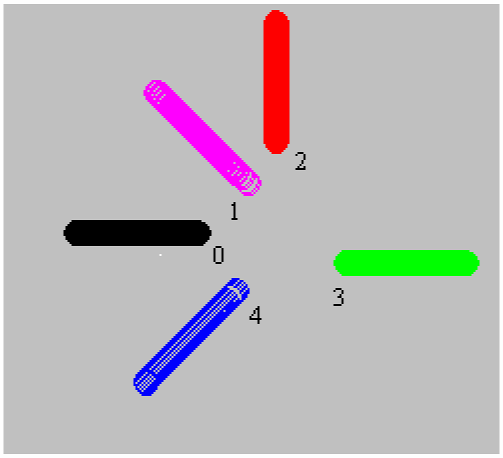
Table 1. The effect of scaling up on the need to cooperate and propagate conflicts

The case of avoiding hidden collisions through benevolence:

Figure 5a depicts a situation where the robots numbered 0 and 1 are in collision. In the normal scheme of collision avoidance robot 1 accelerates and robot 2 decelerates, which however results in collision between robots 0 and 2 in the future as shown in figure 5b. At the time of decision making robot 0 and robot 2 are out of the field of vision of each other and robot 0 is not aware of the consequence of its decision made with respect to robot 1. However robot 1 is in the knowledge of both robot 0 and 2 as they are within its range of vision. Robot 1 can anticipate the future collision between robot 0 and 2 and hence modify the decision making strategy. It could suggest 0 to decelerate while itself accelerates, this would avoid the future collision between 0 and 2. We call this an act of benevolence where 1 comes forward and changes its own preferred mode of action (that of decelerating) to avert collision between 2 and 0.

The effect of scalability:

The effect of scalability on the need to cooperate and propagate conflict was studied. Table 1 depicts the average number of times when cooperation and conflict propagation had to be resorted to in a system that involved large number of robots. For each system involving certain number of robots a number of runs were performed by assigning random starting and goal locations. The average number of conflicts and propagations for each such system is tabulated below. The results suggest that the need to cooperate in a multi-robotic system increase when the system scales up to a large number of robots.



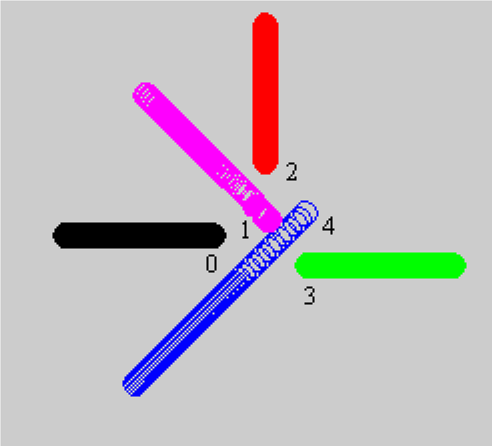


Figure 4a (above left). A snapshot of a system of five robots

Figure 4b (above right). Robots 1 and 3 propagate requests to resolve their conflicts to robot 4, which accepts the request and moves faster such that 1 and 3 are able to avoid their mutual direct conflict.

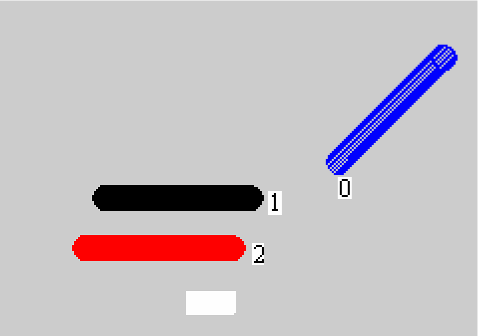


Figure 5a. Robots 1 and 0 detect conflict between each other. Robots 0 and 2 are not aware of each other

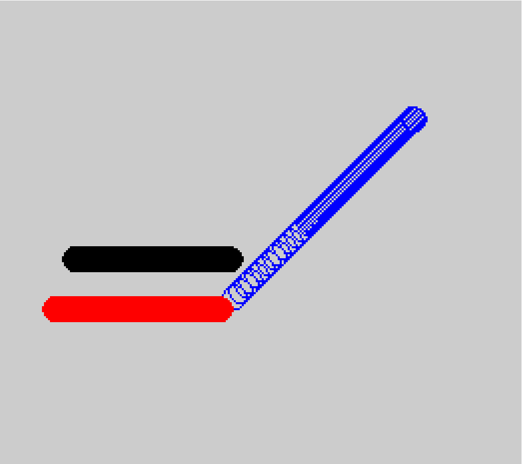


Figure 5b. Robot 0 accelerates and 1 decelerates that results in collision between 0 and 2

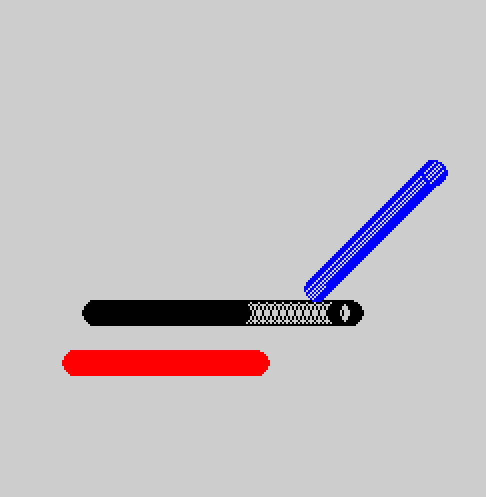


Figure 5c. Collision between 0 and 2 averted through an act of benevolence of 1

4. Application of the Architecture to a Multi-Robot Surveillance System

Multi robot surveillance finds many applications such as in border patrol, beachfront surveillance and reconnaissance of secured rural areas and cities. See Figure 3 for our adapted architecture to this domain. Our abstract surveillance model consists of robots located as shown in Figure 6 along the two main diagonals of the surveillance zone. Intruders enter the zone from one of the four sides and leave via any of the remaining three sides. The robots coordinate to track as many of the intruding targets as possible. The top most loop or the social reasoning loop involves collaborative processes between robots. A robot that first detects a target writes information about the target to a common space, which can be accessed by other robots. The detecting robot also transmits requests specific to certain robots asking for an agreement to track the target in future. A robot that agrees to track in future enters an acceptance relation while a robot that is unable to agree is in a rejection relation with the robot that requests. The overall group need is one of maximizing tracking efficiency or to track as many targets as possible, with each target being tracked for a minimum duration of time. According to these criteria the robot reasons about its autonomy and enters into collaboration and negotiation with other robots. The negotiation process involves the following:

1. Determining the group of recipient robots who would receive information about an intrusion detection
2. The strategy by which a tracking robot stops tracking the intruder and hands over the responsibility to another robot. For example in the simple case where there’s only one intruder as shown in Figure 5a, the robot ‘Ri’ first detects and begins to tail the robot and after a while passes the responsibility to either ‘Rj’ or ‘Rk’. However if ‘Rj’ and ‘Rk’ had been tied to other intruders a negotiation may ensue between the robots depending on the motion direction of the other intruders.
3. The role for social values and norms such as *reciprocity* and *benevolence* in the negotiation process. For example in figure 5b, initially, robot Ra tracks the intruder ‘a’ and robot Rb tracks intruder ‘b’. When ‘b’ changes its direction course Ra offers to track ‘b’ in addition to ‘a’ thereby releasing Rb free to take up other commitments. We say Ra is being benevolent to Rb
4. The utility of shared reasoning in the negotiation process. For example in figure 5c intruder ‘a’ is detected by ‘Rm’ and ‘Rp’ while intruder ‘b’ is detected by ‘Rn’ and ‘Rp’. Since the information about ‘a’ and ‘b’ is available to ‘Rp’ it can make use of this to advise ‘Rm’ not to track ‘a’ but wait for ‘b’ and track ‘b’ and similarly advise ‘Rn’ to wait for ‘a’ and not to track ‘b’.

At the individual level the robot balances its local priorities to track the intruders within its field of vision against the global priorities assigned to all the intruders in the system. The local and global priorities are formulated through a scheme of fuzzy inference technique, which will be reported elsewhere. Based on the balanced priorities assigned to the intruders within its field of vision as well as its own levels of autonomy the robot makes a decision to track a particular target or maintain status quo. The third loop concerns with the consents to track targets in future made by the robots and evaluating whether such consents would be maintained or broken.

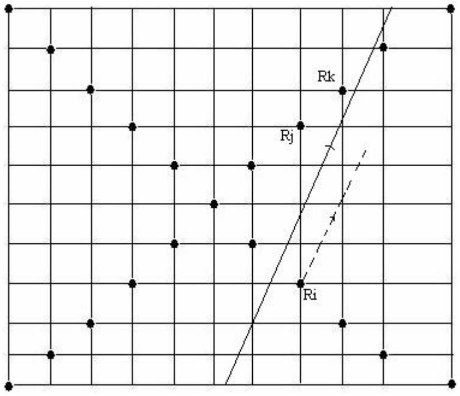


Figure 6a. Initial tracking is through robot ‘Ri’ after which either ‘Rj’ or ‘Rk’ take over depending on other commitments

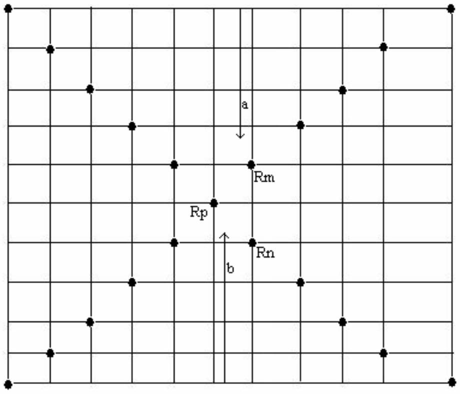


Figure 6b. Shared reasoning involving robots ‘Rm’, ‘Rp’ and ‘Rn’ about intruders ‘a’ and ‘b’

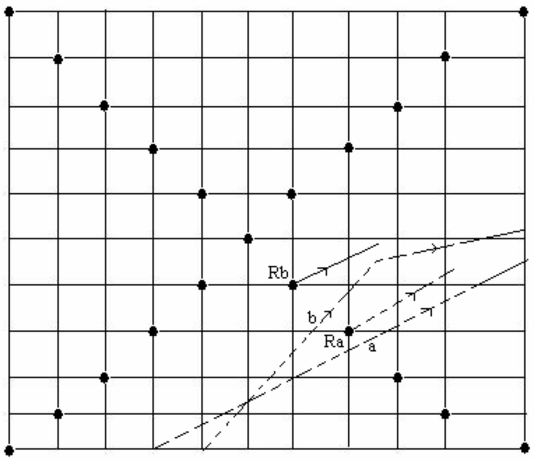


Figure 6c. Ra tracks ‘a’ and Rb tracks ‘b’ initially. When ‘b’ changes track Ra offers to track ‘a’ and ‘b’ leaving Rb free for other commitments

5. Conclusions

We presented an architecture for distributed collaborative and shared reasoning amongst several agents on a surveillance task. The architecture accounts for reasoning about other agents and group membership as well as behavior prediction of other agents. We have begun to implement this architecture and as a first step we have shown nontrivial negotiation to avoid collision among a group of mobile robots. Cooperation becomes more necessary as the number of robots is increased.

Acknowledgements

This work is supported by AFOSR grant F49620-00-1-0302.

References

[1] K. M. Krishna, H. Hexmoor and S. R. Pasupuleti, “Avoiding Collision Log-jams through Cooperation and Conflict Propagation”, in Proceedings of *KIMAS’ 03 (Knowledge Integrated Multi Agent Systems)*, Boston, MA, 2003

[2] K. M. Krishna and H. Hexmoor, “Towards Quantification of Cooperation between Robots”, In Proceedings of *PERMIS’03, Performance Metrics in Intelligent Systems*, Washington DC, 2003

[3] P. Poli, and H. Hexmoor, “Effects of Power on Trust and Autonomy”, In AAMAS-03 workshop titled: Autonomy, Delegation, and Control, Melbourne, Australia, 2003.