A cooperative policy for conflict resolution to multi-agent exploration

Ket-ziquel Hernández G., Abraham Sánchez L. and Francisco Sosa H.

Benemérita Universidad Autónoma de Puebla
Facultad de Ciencias de la Computación
14 Sur y San Claudio, 72570 Puebla-Pue, México
ket2107@hotmail.com, asanchez@cs.buap.mx, quick_falcon@yahoo.com.mx

Abstract. Although most mobile robotic systems use a single robot that only operates in its environment, a number of researchers have considered the advantages and disadvantages of the potential use of a group of robots that cooperate for the accomplishment of a required task. This paper presents a method to explore an unknown environment by multi-agent robots, which is a parallelization of the SRT (Sensor-based Random Tree method). Several coordination strategies to solve the cooperative exploration problem were proposed, in particular we focused our attention in the cooperative policy strategy. This policy is completely decentralized, as each robot decides its own motion by applying some rules only on the locally available information. Simulation results show the practicality of the approach.

Keywords: Multi-agents, exploration, randomized algorithms.

1 Introduction

The advantages of collaborative behavior have been examined extensively in the context of biological systems. In the field of robotics, research on multi-robot systems is gaining popularity because the multi-robot approach provides distinct advantages over a single robots such as scalability, robustness and speed. On the other hand it increases the complexity of the system by adding more parameters into the problem space. Exploration using multiple robots is characterized by techniques that avoid tightly coordinated behavior.

Many studies have examined the utility of communication for the purpose of behavior coordination. Some find that communication is beneficial [1], others conclude that communication does not aid coordination [2]. One possible explanation for these inconclusive findings is that there is not a simple "yes or no" answer to whether communication is beneficial; the answer may be 'yes' for certain agent configurations and 'no' for others. Yet, interactions between sensory range and communication range each can substantially impact the benefit of communication, as can the structure of the environment (e.g., how objects of potential interest are distributed, randomly or otherwise), but their effects are not examined systematically in previous studies. Here, *communication range* refers to the maximum distance across which two agents can communicate, and *sensory range* refers to the maximum distance objects of potential interest can be from the agent and still be detected.

Agent coordination and communication are important issues in designing decentralized agent systems. Since exploration task requires cooperation and coordination among robots, the achievement of the task will be accidental if robots work independently. This task can not be done unless robots cooperate and coordinate their behaviors. Therefore, it is necessary to have cooperation strategies that allow multiple robots to help each other in the problem solving process. The proposed cooperative policy is detailed in Section II. Simulation results in different environments are discussed in Section III. Finally, conclusion and future work are detailed in Section IV.

2 The cooperative policy

Frazzoli et al., proposed a novel policy for steering multiple vehicles [3], the policy rests on the assumption that all agents are cooperating by implementing the same rules. They mentioned that their policy is completely decentralized, as each robot decides its own motion by applying those rules only on the locally available information processed by each robot. Their policy applies to systems in which new mobile robots may enter the scene and start interacting with existing ones at any time, while others may leave. Each agent enters the environment at initial configuration, and is assigned a goal configuration. The agents move along a continuous path. A conflict is said to occur at time t_{conf} between the i-th and the j-th agents, if the agents are closer than a specified safety distance.

The proposed spatially decentralized control policy is based on a number of discrete modes of operation. Next the operation modes of the policy are described and later the properties.

- Reserved region: This policy is based on a concept of reserved region, over which active robot claims exclusive ownership.
- Constraints: A sufficient condition to ensure safety is that the interiors of the reserved regions are disjoint at all times; if such a condition is met, conflicts can be avoided if robots hold their reserved fixed regions, and move within them.
- Holding: The robot can be stopped at any time, by setting the bounded signed curvature. One can say that the robot is in the hold state.
- Right-turn-only steering policy: The concept for decentralized conflict-free coordination is based on maintaining the interiors of reserved regions disjoint.
- Rolling on a stationary neighboring reserved region: If the path of the reserved region to its position at the target is blocked by another reserved region, a possible course of action is represented by rolling in a pre-specified direction.
- Nonstationary neighbors: The reserved regions of a robot will not necessarily remain stationary while a robot is rolling on it.

– Generalized roundabout policy: The policy followed by each mobile robot is based on four distinct modes of operation, each assigning a constant value to the control input. The behavior of an individual robot can be modeled as a hybrid system (see [3] for more details).

The problem of certifying the admissibility of a requested plan can de dealt with most effectively by decoupling the safety and liveness aspects of current and final configurations. Indeed, for a given policy ξ, one can consider the following two properties.

$\mathbf{P_1}$: A configuration set $G = \{g_i, i = 1, ..., n\}$ is unsafe for the policy ξ if there exist a set of target configurations $G_f = \{g_f, i = 1, ..., n\}$ such that application ξ leads to a collision.
$\mathbf{P_2}$: A target configuration set $\{g_f, i = 1, ..., n\}$is blocking for the policy ξ if there exist a set of configurations $G = \{g_i, i = 1, ..., n\}$ from which the application of ξ leads to a dead-lock or live-lock.

A plan $(G(t), G_f)$ is admissible if it verifies the predicate $\neg\mathbf{P_1}(G(t))\wedge\neg\mathbf{P_2}(G_f)$. Simple tests to check both properties are needed for the generalized roundabout policy.

We can mention the following properties of the proposed policy, which can guarantee to the system to be collision-free.

– Admissibility. One can consider a framework in which new robots may issue a request to enter the scenario at an arbitrary time and with an arbitrary 'plan', consisting of an start and goal configuration. It is important to have conditions to efficiently decide on the acceptability of a new request, in other words, whether the new proposed plan is compatible with safety and liveness of the overall system.
– Well-posedness. One can verify that the generalized roundabout policy leads to a well-posed dynamical system, i.e., a solution exists and is unique, for all initial conditions within a given set.
– Safety. Within each state, the feedback control policy has been chosen so that reserved discs (a geometrical form created from its sensors) never overlap: a transition is always enabled to the hold state, which stops the reserved disk instantaneously. Since the robots are always contained within their reserved disk, at a certain distance from its boundary, safety is ensured.
– Liveness. It is necessary the definition of a condition concerning the separation of reserved discs associated with target configurations. In other words, any circle of radius $\rho(m)$, with $1 < m \leq n$, can contains at most $m - 1$ reserved disk centers of targets.

3 Simulation results

The SRT method, is an exploration method based on the random generation of robot configurations within the local safe area detected by the sensors [4], [5].

A data structure is created, which represents a roadmap of the explored area with an associated safe region (SR). The design of the cooperative exploration strategy proceeds from the parallelization of the basic SRT method, each robot builds one or more partial maps of the environment, organized in a collection of SRTs [6]. Each node of an SRT represents a configuration which was visited by at least one robot, together with the associated local safe region. An arc between two nodes represents a collision-free path. The tree is incrementally built by extending the structure in the most promising direction via a biased random mechanism. The presence of other robots in the vicinity is taken into account at this stage in order to maximize the information gain and guarantee collision avoidance [7].

Consider a population of n identical robots. Each robot is equipped with a ring of range finder sensor or a laser range finder, the sensory system provides the local safe region $S(q)$. The robots move in a planar workspace, i.e., $\mathbb{R}^2$ or a connected subset of it; the assumption of planar workspace is not restrictive, 3D worlds are admissible as long as the sensory system allow the reconstruction of a planar LSR for planning the robot motion. Each robot is a polygon[1] or another shape subject to non-holomic constraints. The robot also knows its configuration q, one can eliminate this assumption by incorporating a localization module in the method. The robots know its ID number and each robot can broadcast within a communication range R_c the information stored in its memory (or relevant portions of it) at any time. The robot ID number is included in the heading of any transmission. The robot is always open for receiving communication from other robots inside R_c.

The exploration algorithm for each robot is shown in Figure 1. First, the procedure BUILD_SRT is executed, i.e., each robot builds its own SRT, $\mathcal{T}$ is rooted at its starting configuration q_{init}. This procedure terminates when the robot can not further expand $\mathcal{T}$. Later, the robot executes the SUPPORT_OTHERS procedure, this action contributes to the expansion of the SRTs that have been built by others robots. When this procedure finishes, the robot returns to the root of its own tree and finishes its exploration. For more details of the approach, one can consult the work [7].

BUILD Multi-SRT(q_{init})
1 $\mathcal{T}.init(q_{init})$
2 BUILD_SRT($q_{init}.\mathcal{T}$);
3 SUPPORT_OTHERS(q_{init});

Fig. 1. The Multi-SRT algorithm.

[1] Polygonal models make it possible to efficiently compute geometric properties, such as areas and visibility regions.

Figure 2 shows two different views of the execution of the Multi-SRT algorithm in an environment that contains 5 nonholonomic mobile robots. The robots are initially grouped in a cluster. The environment is a square region with a garden-like layout, where each area can be reached from different access points. The second environment used for the experimental part is also a square, it contains some obstacles of different shapes.

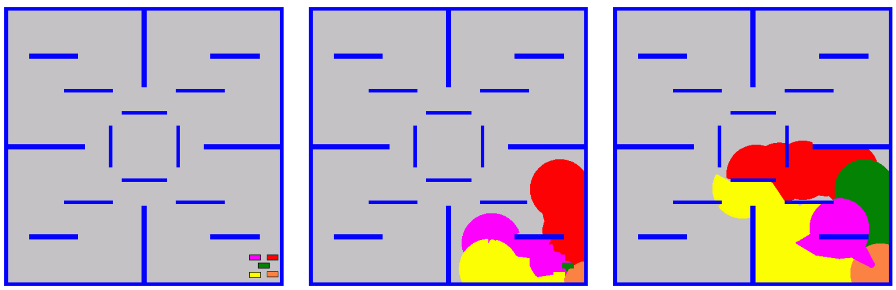

Fig. 2. Snapshots showing the execution of the Multi-SRT algorithm.

In this work, several strategies were utilized to solve multi-robot exploration tasks. The two originally proposed strategies in the work presented in [7] were also considered (i.e., coordination via arbitration and coordination through replanning). A blackboard system in general, is a distributed, opportunistic approach to system design. It is characterized by a set of knowledge sources that can communicate with each other via an area of global memory called a blackboard. Each knowledge source is designed to solve a specific component of the problem that the system is presented with. From a behavior-based prospective, these knowledge sources are generally represented as individual behaviors. One of the key components of any blackboard system is the arbitration mechanism, which is the component of the system that coordinates behavior execution.

The tests were performed on a Celeron © 430 processor-based PC running at 1.80 Ghz with 2 GB RAM. The strategies were implemented in Visual C++, taking advantage of the MSL library's structure and its graphical interface[2]. The GPC library developed by Alan Murta was used to simulate the sensors perception systems[3]. The polygonal representation facilitates the use of the GPC library for the perception algorithm's simulation. Figures 3 and 4 illustrate the Multi-SRT and explored regions with clustered and scattered starts respectively. We can see the difference when the robots are evenly distributed at the start of are clustered. At the end of the exploration process, the environment has been completely explored and the SRTs have been built. In these figures, one can observe that each robot built its own SRT and when one of them finished, this entered the support others phase.

[2] http://msl.cs.uiuc.edu/msl/

[3] http://www.cs.man.ac.uk/∼toby/alan/software/

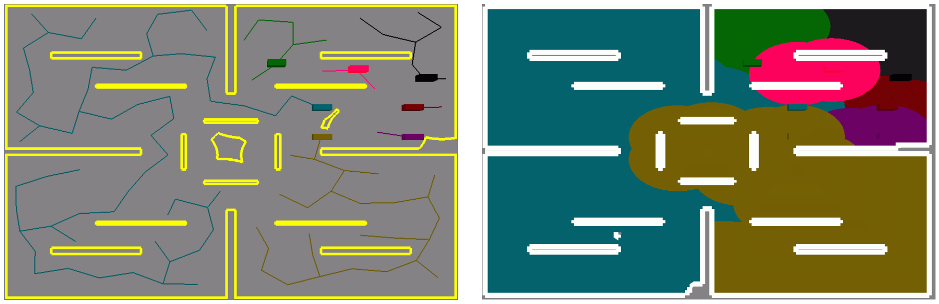

Fig. 3. The Multi-SRT and explored regions with clustered starts with a team of 7 robots.

Exploration time for teams of different cardinality are shown in Figures 5 and 6, both in the clustered and scattered starts. The graphics generated with the exploration times, show that for those environments with a scatter initial configuration, the communication strategies more efficient are the limited communication with messages and the limited communication with no messages. The disadvantage of the limited communication without messages strategy, is that it will require more re-exploration in the presence of more robots in the environment, increasing the exploration time, unlike the limited communication with messages that prevents re-exploration. In the case of the initial cluster configuration, it can be seen that, as more sophisticated and less centralized is the communication between the robots, as more optimal are the results corresponding to the exploration and the conflicts resolution.

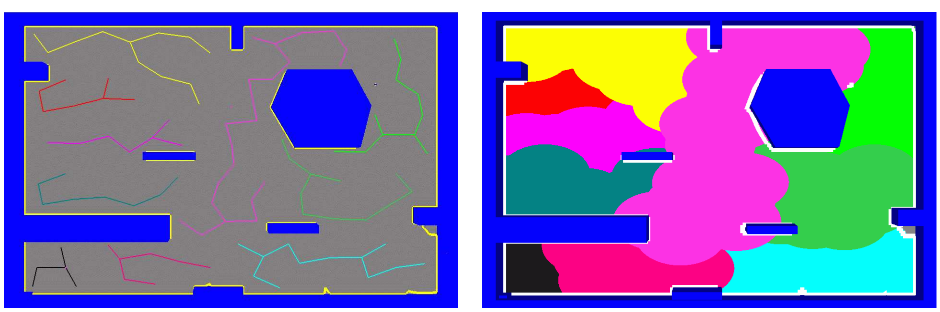

Fig. 4. The Multi-SRT and explored regions with scattered starts with a team of 10 robots.

The analysis of the results obtained for the case of the initial configuration in cluster shows that any communication strategy either limited or unlimited is useful for resolving disputes related to collisions and to carry out the exploration efficiently. The communication strategy with blackboard tends to be sluggish when the number of robots in the environment is increased, because the structure used as a board is shared and only one robot at a time can access it; if more than a robot wants to access the board at the same time, it must wait in line for a turn.

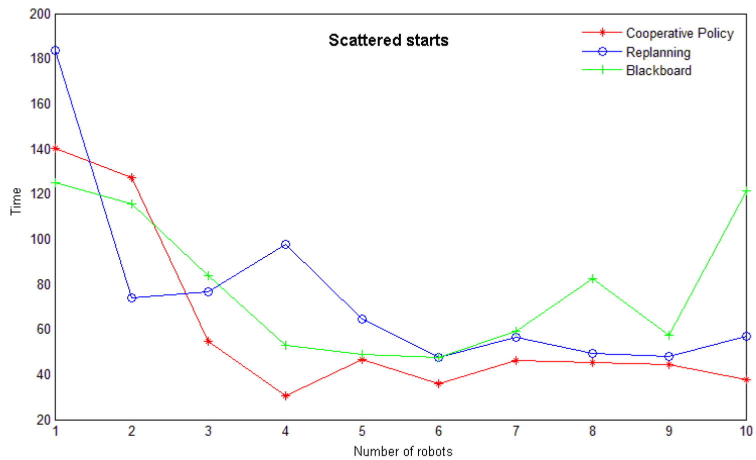

Fig. 5. Environment 2 exploration with scattered starts.

The cooperative policy shows the best results, even in those environments considered difficult to explore. From a particular point of view the good performance of this policy is because, it initially performs a quick exploration of the environment. This can be verified after analyzing the implementation of our approach with all the coordination strategies, and compare their performance in the graphics obtained with the cooperative policy strategy, in the relief phase that the robots use to complement the exploration.

Single robots can not produce accurate maps like multi-robots. The only advantage of using a single robot is the minimization of the repeated coverage. However, even though repeated coverage among the robots decreases the mission's efficiency, some amount of repeated coverage is a desirable situation for better efficiency. Additionally, this better efficiency can be achieved by coordination among robots. The two coordination strategies (the cooperative policy and the coordination through replanning) were compared through simulations. We used the same environment to prove the efficiency of the cooperative policy over the replanning strategy and the same free parameters, see Figure 7.

3.1 Discussion

An appropriate task for the study of communication's utility will require agents to work directly together; if agents can complete the task individually without ever interacting with others, it will be unsurprising to find that communication

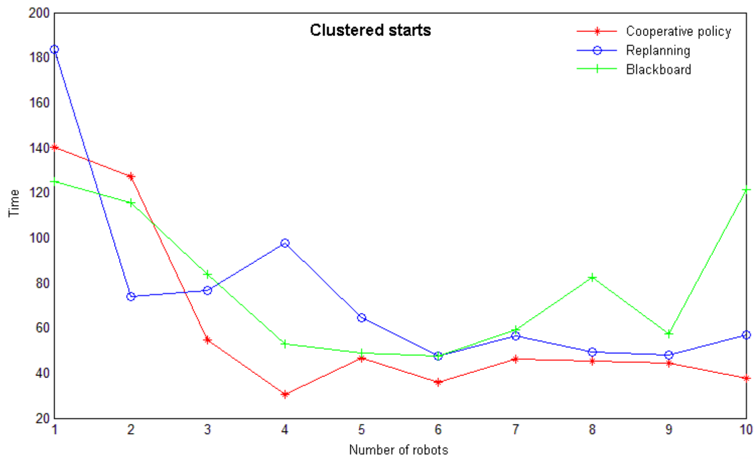

Fig. 6. Environment 2 exploration with clustered starts.

is not useful, whereas if they require the aid of others (even if it is only to improve performance), communication may be adaptive. Ideally, the task should be parameterized to allow the degree of reliance on other agents (and hence the amount of coordination required) to be varied, making it possible to probe the limits of communications utility (e.g., what is the lowest coordination requirement at which communication proves beneficial?). However, care must be taken to ensure that the task is not tailored too specifically to the search for communication.

The multi-agent and multi-robot research communities have each developed their own methods for perception, reasoning, and action in individual agents (or robots). Therefore, at the level of explicit coordination among multiple individuals, the differences between techniques used in multi-agent systems (MAS) and those used in multi-robot systems (MRS) are in fact very few. This is not to say that MAS and MRS are equivalent in any fundamental way, but rather that although robotics researchers employ sophisticated specialized techniques of various sorts (e.g., control theoretic, probabilistic among others) when designing single-robot control systems, they have so far tended to use techniques that are already well-known in the agent community when designing explicitly coordinated MRS.

It remains an open question as to how much benefit can be derived from using sophisticated coordination methods in MRS, because of an important underlying issue, which we suggest is one of the primary challenges facing MRS (and MAS): utility. In any case, since coordination is achieved by maximizing utility (or, equivalently, minimizing cost), the utility measure must account for all state information that is relevant to the task. All information that affects task performance but is not captured in the utility measure is captured in what economists refer to as externalities, the effects of which can be disastrous.

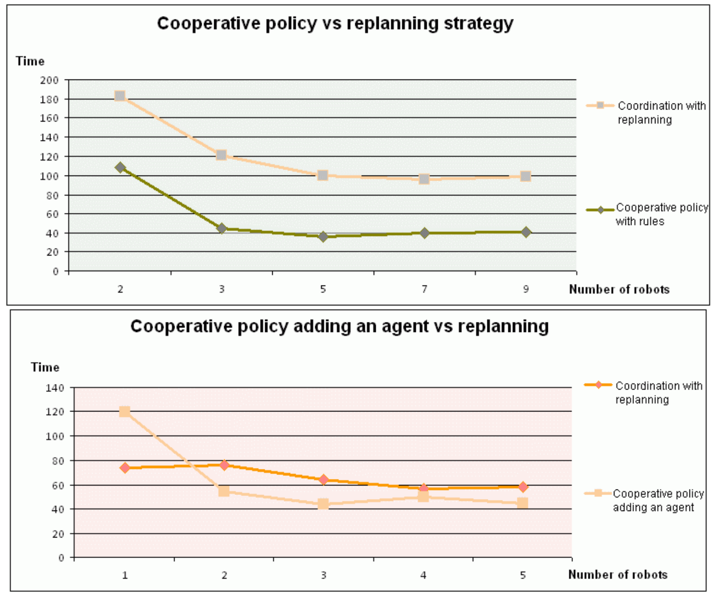

Fig. 7. The decentralized cooperative policy vs. replanning.

4 Conclusions and future work

One of the fundamental challenges in mobile robotics is to explore the unknown environment efficiently and effectively. The efficiency and effectiveness of the exploration are typically measured by the map coverage, map accuracy and exploration time. To efficiently explore an unknown environment with a team of robots, a coordinated strategy that maximizes the exploration area is required. Exploration of an unknown environment is one of the major applications of Multi-Robot Systems. Many works have proposed multi-robot coordination algorithms to accomplish exploration missions based on multi-agent systems techniques.

The implemented policy gives rise to a hybrid system, which can be shown to be well posed and safe, if the initial configurations satisfy a rather nonrestrictive condition. Through the examples, we can affirm that the policy is spatially decentralized and its complexity is bounded regardless of the number of agents. It can be concluded that the integration of a communication strategy in the robots, as a way of coordination to avoid conflicts, is very useful in the task of environment exploration, because it can help them to share information in order to avoid conflicts related to collisions and, in some cases, to prevent re-exploration areas in the most important phase, when a good communication strategy can contribute in reducing the exploration time.

Exploration and localization are two of capabilities necessary for mobile robots to navigate robustly in unknown environments. A robot needs to explore in order to learn the structure of the world, and a robot needs to know its own location in order to make use of its acquired spatial information. However, a problem arises with the integration of exploration and localization simultaneously. The integration of a localization module into the exploration process based on SLAM techniques will be an interesting topic for a future research. We can also consider an extension of the Multi-SRT exploration method, where the robots constantly maintain a distributed network structure.

References

1. D. Marocco, A. Cangelosi and S. Nolfi, "The emergence of communication in evolutionary robots", *Philosophical Transactions: Mathematical, Physical and Engineering Sciences*, Vol. 361, No. 1811, (2003) 2397-2421
2. P. Grim, T. Kokalis, A. Tafti and N. Kilb, "Evolution of communication with a spatialized genetic algorithm", *Evolution of Communication*, Vol. 3, No. 2, (2002) 105-134
3. E. Frazzoli, L. Pallotino, V. G. Scordio and A. Bicchi, "Decentralized cooperative conflict resolution for multiple nonholonomic vehicles", *Proc. of the American Institute of Aeronautics and Astronautics Conf.*, (2005)
4. G. Oriolo, M. Vendittelli, L. Freda and G. Troso, "The SRT method: Randomized strategies for exploration", *IEEE Int. Conf. on Robotics and Automation*, (2004) 4688-4694
5. J. Espinoza L., A. Sánchez L. and M. Osorio L., "Exploring unknown environments with mobile robots using SRT-Radial", *IEEE Int. Conf. on Intelligent Robots and Systems*, (2007) 2089-2094
6. A. Franchi, L. Freda, L. Marchionni, G. Oriolo and M. Vendittelli, "Decentralized cooperative exploration: Implementation and experiments", *Intelligent Autonomous Systems 10*, (2008)
7. A. Toriz P., A. Sánchez L. and M. A. Osorio, "Coordinated multi-robot exploration with SRT-Radial", *LNAI 5290*, (2008) 402-411