Guiding and Regrouping People Missions in Urban Areas Using Cooperative Multi-Robot Task Allocation.

Anaís Garrell
Institut de Robòtica i Informàtica Industrial
IRI (UPC-CSIC), Barcelona, Spain
agarrell@iri.upc.edu

Oscar Sandoval-Torres
Institut de Robòtica i Informàtica Industrial
IRI (UPC-CSIC), Barcelona, Spain
osandoval@iri.upc.edu

Josep Maria Mirats-Tur
Centro tecnólgico del agua, Cetaqua
Barcelona, Spain
jmirats@cetaqua.com

Alberto Sanfeliu
Institut de Robòtica i Informàtica Industrial
IRI (UPC-CSIC), Barcelona, Spain
sanfeliu@iri.upc.edu

Abstract

This article presents a novel approach for solving people guidance in urban settings suported in Multi-Robot Task Allocation. The developed architecture overcomes some of the limitations of existing approaches, which are either tailored to tightly bounded environments, or based on unrealistic human behaviors. In particular we define a "Selfish Task Allocation", the novelty of this proposal is the ability of robots to naturally cooperate if they need to do so, without the need to pre-set the interaction between them by an operator. Some simulated experiments about people guidance where robots are able to respond to real situations are presented; the failure of some robots, the group spliting up, people leaving the group, the addition of new elements to the team or the appearance of new tasks (lead other groups) are some of the situations being considered.

1. Introduction

In recent years, interest of researchers on social robots and cooperative robotics has increased significantly. The applications of this field are very diverse, some examples are exploration sites [29], robot formation for navigation [11] or object transport and evacuation of people [7]. Researchers soon wondered which labours may be tackled by a group of robots working cooperatively and how the robots should behave to solve those tasks. This paper is focused on this second question: How a group of robots should act to solve a task or set of tasks in a cooperative way? More concretely, we focus our research in the problem of guiding a group of people using Multi-Robot teams. This paper solves this problem presented with Multi Robot Task Allocation (MRTA). Dealing with the MRTA problem one must also address many of the other issues involved in multi-robot systems: communication, conflict resolution or information sharing, among others.

In this present work, a new orientation is presented, where the main question is not about the division of tasks between robots. In the developed approach the participation to solve a task is not limited to a single robot. Robots will try to participate in the tasks that give them more benefits, even when the task is already being done by someone else. In many cases the tasks can be performed by more than one robot, and even more, it is probable that the task is accomplished quicker or ”better if many robots are involved. This feature has not been explored so far by other existing architectures and we present here a first draft. We shall apply our MRTA proposal to the guiding people challenge in urban areas. A set of experiments in which a set of robots must cooperate carrying out various tasks of people guidance are presented.

Here, we will consider a group of robots where each robot executes a kind of task, which can be exchanged according to environment situation. One of the robots is the *leader*, as a human tour-guide. It is placed at the front of the group and its role is to estimate the trajectory of both the people and the rest of robots. The other robots, called *shepherds*, are responsible for guiding the people, preventing any person leaving the group, and following the path given by the leader.

Obstacles of the environment, such as buildings or benches, are considered through a potential field, where the positions of people and robots are represented by continuous and derivable functions. Since the obstacles are assumed to be static, their positions are represented by constant functions. Using these parameterizations each point in the space will have assigned a potential value, which will be used for the navigation of robots toward the environment.

This paper contents has been distributed as follows.

Figure 1. A group of people being guided by a set of robots.

We start by discussing related work of people guidance and task allocation. Section III describes the model we used for people guidance and we define which tasks robots must solve cooperatively. Section IV present a novel proposal of Task Allocation. In section V the experiments and results are presented, and last but not least, in section VI some conclusions and future directions are provided.

2 RELATED WORK

Developing social and cooperative robots is a novel field within robotics. Consequently, if we refer to the challenge of guiding a group of people in urban areas the number of related reference is not very large. There has been some research using a single robot for guiding people in exhibitions and museums [5], in hospitals or as an assistant [10]. Nevertheless, the main purpose of these robots was simply educational or to entertain, instead of guiding people. Similar applications have been developed for evacuating emergency areas, detecting hazardous materials, or offering task assistance to humans. Animal flocks were automatically controlled using a single robot in [3, 21]. Again, the cooperative behavior of our approach is not exploited in these methods. Furthermore, the environments where the systems are shown to work are highly controlled, do not include obstacles and are tightly closed.

The methods mentioned previously consider either single robots, or multiple robots moving independently from the rest. To our knowledge, only a few works deal with multiple robots behaving in a cooperative mode. For instance, [14] performs a qualitative analysis of the movements of different entities, humans or animals, and it builds an architecture of three robots to guide them. However, realistic situations, such as obstacles or dealing with people leaving the group are not considered. In [17] several types of robot formations, and different strategies for approaching the robots to the people are considered. However all these issues and the general movements of the robots are ruled by a large number of heuristics which makes the system impractical. Furthermore, in order to achieve the desired guiding results, robot motions with almost infinite accelerations are required.

As we have mentioned above, for solving the problem of guiding groups of people with multiple robots, we will use a new architecture of MRTA. The MRTA problem has recently become a key research topic in Multi-Robot Systems. “It deals with the way to distribute tasks among the robots and requires to define some metrics to assess the relevance of assigning given tasks to such or such robot” [30]. There are many approaches to MRTA. Some of them tackle the problem with a centralized approach [4, 6]. The main advantage of these approaches is they can offer an optimal solution. However, they have several disadvantages including intractable solutions for large teams, slow performance, communication dependence, and that the central station become a crucial point of failure.

The most popular way to tackle MRTA are distributed architectures, they have a completely different orientation compared to centralized ones. There are different classes within distributed architectures. First, we found the behavior-based architectures as ALLIANCE [18, 19, 20] or Broadcast for Local Eligibility (BLE) [31]. Such architectures have the advantage of modular programming but the existence of cross-inhibition avoid robots working on the same task; therefore, task-cooperation only appears if it is predefined in the robot program. A second class are the market-based architectures, which began in software agents with the Contract Net Protocol [24]. We can find today many adaptations for robotics; some of the most relevant are M+ [2], MURDOCH [16], TraderBots [12, 13] or SET [30]. These architectures mainly work with auctions assigning the task to the most capable robot. Each variant achieves different improvements but, in general, auction-based architectures offer the advantage of ease of adaptation to team size changes. Disadvantages of these kind of architectures are again the difficulty to allow cooperation for solving tasks; some of the architectures allow cooperation in case of error, but is not a constant in normal conditions.

There are some other distributed approaches as role-based [8, 25], swarm intelligence [32] or heuristic search [23]. One of the newest architecture is ASyMTRe [26, 27, 28], representing a task synthesis approach inspired on information invariants implemented by Tang and Parker.

In general, existing proposals attempt to allocate one task to one robot. Only a few consider the case for cooperating to solve a given task, and with the exception of ASyMTRe in which cooperation is somehow preset on the system definition, cooperation only occurs when there is an error during the group operation.

In the present work, we use the advantages of the

MRTA for the distribution of tasks and the cooperation between different robots to solve the particular problem of guiding groups of people in urban areas.

3 People Guidance Method

In previous work, we have presented the "Discrete Time Motion" model (DTM) [15], with DTM robots are able to modelize the representation of the whole environment, made of an open and not bounded area with obstacles, and how the elements of this environment are related with the group of robots and people.

The DTM model, on one hand, estimates position, orientation and velocity of the robots and people, and the position of the obstacles at a time instance k. It will be used to estimate the intersection of the people with the obstacles and detect if someone is leaving the group with a Particle Filter [1]. On the other hand, orientation and velocity of people and robots between two time instances k and $k+p$. It will be used to compute the robots' trajectory to reach the goal while preventing people leaving the group.

The DTM model aims to represent the areas where the robots will be allowed to move, by means of potential fields. In order to decide the trajectories the robots will follow we will define a potential field over the working area, and perform path planning in it [**?**]. These repulsive forces may be interpreted as continuous probability functions over the entire space. Once they are defined, the tensions at each point of the space may be computed as the intersection of these Gaussians.

Having defined the tensions for each of the components of the environment –i.e. robots, persons and obstacles– we are ready to define the potential field. This is computed as the intersection of all the Gaussian functions for a given variances. Once the potential field is known, we will define the trajectories of the robots, based on the position of the persons and the goal and following the paths with minimum energy in the potential field.

The novel proposed MRTA, described below, is applied to this problem of robot cooperation. This work not only studies the interaction between robots and humans, but also investigates the behavior of robots for the accomplishment of the task in a cooperative way, i.e., how robots should distribute themselves for different tasks and how they can manage problems that can suddenly arise while guiding people.

The main tasks that can be found in the guidance are: Tour guide task, shepherd task and recovering people task. In the following subsections we will explain each of the tasks and how they should be developed by robots.

The task of *leading the group* is essential for the positive development of the mission. One of the robots takes the role of leader, as a tour guide, must compute the route that the group should follow and should be placed at the head of the formation. Moreover, it has to solve the path planning issue and be sure that the group is following him.

Algorithm 1 General strategy for guiding people

```
1: Obtain the start point and the goal point.
2: Compute the roadmap with the path planning.
3: Search the shortest path of the roadmap.
4: Mark every node of the shortest path as a subgoal.
5: for Every subgoal do
6:    Act upon the situation (open path, narrow passages...) carrying out the priorities.
7:    Move to the next subgoal
8: end for
```

Robots collaborating in guidance people task which are not leaders are called *shepherd robots*. They must be placed around the group of people to prevent the group spreading away or to prevent individuals escape from the group. Robots perform a repulsion force on people which prevent them to escape in areas near to robot's position. The shepherding task is performed by all the robots except the *leader*, that only carries out the function of a guide. The rest of robots follow the strategy depicted in Algorithm 1. Note that this algorithm does not explicitly consider safety conditions for the persons; i.e, when the robots are working it is necessary to satisfy a set of priorities for the safety of the people, such as avoiding collisions. However this was already taken into account when we defined the security areas in the Gaussian functions parameterizing the tensions.

The formation of robots being considered here is the following: equidistant positions between the robots in the formation in case there are no obstacles, if they exist, obstacles made a repulsive forces on people and therefore robots can use this situation to cover bigger areas around people. See figure2 for an schematic example. Mainly, if there are no additional tasks, the robots played the present task in the course of the mission. A task that can change the roles of shepherd robot is the rescuing people task, which occurs when one or more individuals distance themselves from the training, which is explained below.

A common problem in people guidance is when a person or a set of people are escaping from the group, at this moment, *recovering people task* is activated. One or more robots must move toward the person who is moving away and, once the robot arrives to the escaping person position, it must accompany him/her back to the formation. To solve this problem robots must estimate people's location by using, for instance, a particle filter [1], and once an estimation of future position is known, they must compute the new trajectory using a path execution module. Once the individual is intercepted, the robot must accompany him back to the group. While one of the robots perform this task, the rest of shepherding robots must recalculate their positions so the new formation covers the largest possible area. In this new vision of task allocation, robots performing this task could be robots which were not initially in the training.

In the next section we proceed to the description of the

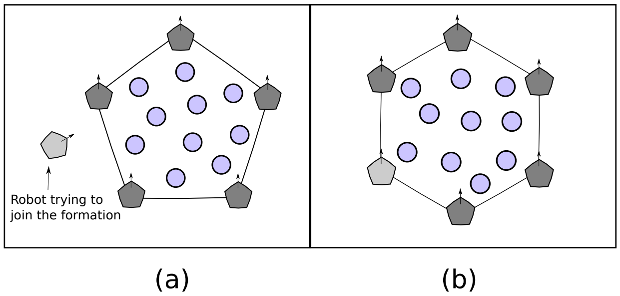

Figure 2. (a) A group of five robots moves in equidistant formation and a 6th robot tries to join the group, (b) the sixth robot has been introduced in the formation and the group has rearranged their positions to maintain equidistant training.

new architecture of MRTA and how it is applied to the development and distribution of tasks presented in this section.

4 Selfish Task Allocation

In our MRTA proposal, participation in the task solving is not limited to a single robot. Robots try to solve those tasks providing them more benefits, even if the task is already being done by someone else. Often tasks can be performed by more than one robot, and it is even probable that the task is solved faster or ”better if many robots are involved. This section first describes this novel approach for MRTA called *Selfish Task Allocation*, and then its application to the problem of people guidance using multiple robots.

A MRTA algorithm distributes tasks among robots, thus, it is necessary to define some metrics to assess the relevance of assigning given tasks to particular robots. Up to date literature MRTA architectures address the problem trying to assign one robot to one specific task. Nevertheless, if there is a task that needs to be tackled by more than one robot, these architectures divide (usually in a manual fashion) this particular task in smaller subtasks and then introduce those individual chunks in the system.

The presented Selfish Task Allocation (STA) is able to address those tasks requiring more than one robot. It improves the overall system performance when there are more robots than tasks without losing system decentralization, and maintains a fault control system. Furthermore, to achieve this features, we addressed the problem from the perspective of being the robots who define which tasks they perform better (hence the name *selfish*). In this way, with STA cooperation emerges naturally from within the system while existing approaches go the other way, tasks are divided between the robots and hence real cooperation is never achieved although pre-set in the system programming.

The architecture shown in this work was inspired by human behavior, because a person normally performs the task that gives more benefits, without knowledge about the details of other group members (mood, ability to perform tasks, etc), and even so, without the knowledge of the whole system the tasks are solved. With regard to implementation, our architecture was inspired by the one presented by Tang and Parker in [28], called ASyMTRe. STA contains a set of schemas, which represent different capabilities of robots. Such schemas are commonly composed of information necessary for its execution (inputs) and resulting information (outputs). However, our proposal has some differences with ASyMTRe, the first is that our proposal is presented as a online Task Allocation, while ASyMTRe is an offline system, second in ASyMTRe robots cooperate only when is neccesary, in our proposal robots cooperate in all the cases where the system performance is increased, third ASyMTRe requires that all the system tasks are resolved, if they can solve most of them but not all, then ASyMTRe believes that the system has no solution, our implementation considers that there are some pending tasks, that wait for system or environmental changes, while other tasks are carried out. At least but not at last, the utility function of ASyMTRe is based on cost and probability of each schema, and that information is predefined by the designer, we replace probability and cost by uncertainty, that is a concept with much more mathematical support and also closest to robotics.

The STA model maintains, in a distributed database, a list with the different tasks that robots must solve (system tasks), each task within the list may correspond to a schema in some robot, that schema needs to fullfill their inputs to be activated, inputs comes from outputs of other schemas and therefore there are several ways (combinations) to solve a task. Every schema has associated an implementation uncertainty value U_i, that value is determined by a Gaussian distribution, hence its uncertainty is defined by the covariance function. Therefore, the uncertainty U_i is computed from a sampling method. As mentioned, to activate a schema, all their inputs must be connected, then if some input is not achieved, the uncertainty of the schema is increased thus, the schema uncertainty U_e can be written as:

$$U_e = U_i exp^{Penalty} \tag{1}$$

Where,

$$\begin{aligned} U_i &= \text{Implementation Uncertainity} \\ Penalty &= \text{Number of Inputs not achieved} \end{aligned}$$

Penalty represents the number of inputs in the schema which have not been already achieved, for instance, if in a box pushing task the robot does not have a schema to solve auto-localization the $Penalty = 1$.

In addition, once the uncertainty function of each schema is known, it is required to compute the uncertainty function for an specific task. Each task consists on a set ε, whose elements are the necessary and sufficient schemas to solve the task. The task uncertainty is defined by:

$$U_T = U(v, \Delta x) + (NR+1) \sum_{e \in \varepsilon} U_e \tag{2}$$

Where,

$$\begin{aligned} NR &= \text{Number of robots developing the task} \\ U(v,\Delta x) &= \text{Localization Uncertainty} \\ \Delta x &= \text{Total travel distance for task resolution} \\ v &= \textit{velocity} \end{aligned}$$

It is worth to mention that the set ε is not unique, there are different schema combinations which can solve the task. Hence, we will consider the set ε such that minimizes Eq.2.

The STA considers the presented formulation to obtain a proper task distribution for a set of robots. In following sections, we will apply this theory for solving the problem: “Guiding a group of people in dynamic environments”.

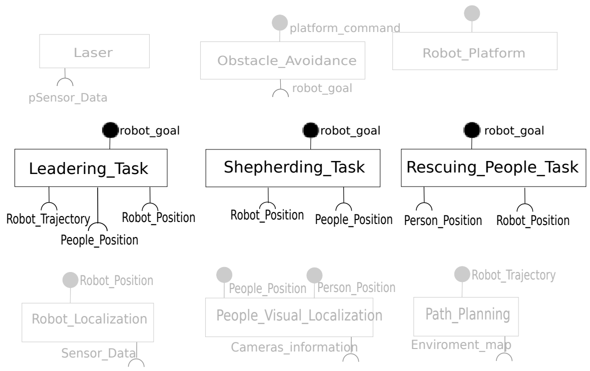

Figure 3. It shows an example of schemas that may have a robot.

4.1 Selfish Task Allocation Applied to People Guidance

In previous subsections we described, on one hand, the architecture of the proposed new model “Selfish Task Allocation”, and, on the other hand, the tasks to be solved in this present work. As explained above, there are three basic tasks in the resolution of the problem of guiding people: Leadering Task, Shepherd Task and People Recovering Task. Each robot has an associated set of schemas, which represent their abilities, in this particular case, each robot has, among others, some or all of these three skills necessary for people guidance.

Each of these tasks are represented by an schema (fig.3), which, being part of the robots, are placed at the disposal of the task allocation for the resolution of possible new tasks, and as any schema, specific people guidance schemas need to complete all their entries.

Firstly, leadering task is the responsible of guide people to their goal, to achieve this objective, this schema needs its own localization, this is a really common input in schemas of mobile robotics because for any interaction (with environment, people or other robots) is necessary to estimate its own position. This schema also needs as input a path, frequently provided by a path planner, that indicates a free route to the group goal, the third input of this schema is the result of their interaction with people, to maintain a close contact with the guided group the robot needs to know people position estimation, which, commonly, are obtained from complex sensor networks, or sophisticated vision software.

Secondly, shepherd task has the function to support the work of leader, keeping the group together. The inputs of this capacity are localization and people position estimate. This last input is used together with team information (team size and id in team) to define the best position for the robot to promote the group integrity.

The last specific schema for people guidance is person recovery, it is the last barrier of guiding people, trying to recover people who are distracted and have been separated from the group. This schema has as entries the position of the distracted person, his own position, and the positions of the group to return the person to the formation.

People guidance schemas are usually part of an extensive set of schemas (as represented in fig.3) that describe the capabilities of a robot, in some occasions the requirements of these schemas are within the robot, and in other cases such inputs have to be looked beside himself.

With the architecture described throughout this paper we have drawn up a series of simulations where we study the behavior of the robots when the administration of the team behavior is delegated to the task allocation, specifically to selfish task allocation.

5 EXPERIMENTS

As mentioned earlier the aim of this paper is to guide a group of people with the help of a group of robots working in a cooperative manner, supported by the selfish task allocation proposal explained previously.

The current work is done within the framework of the European Project URUS [22]. The urban area of work that has been considered for the tests is the North Campus of the Universidad Politecnica de Catalunya (UPC), the Barcelona Robot Lab, which size is about 10000 m^2, see Fig. 4. We will take into account static obstacle as buildings, banks or pots, and dynamic obstacles as mobile robots and people. In an initial state will not be taken into account cars or trucks to simplify the problem.

We use as middleware robotYARP [9], that provides in-

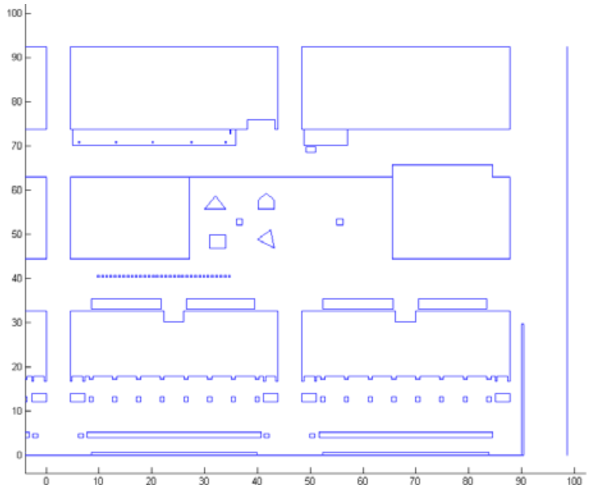

Figure 4. Map of the Campus Nord of Univertitat Politecnica de Catalunya (UPC), Barcelona Robot Lab.

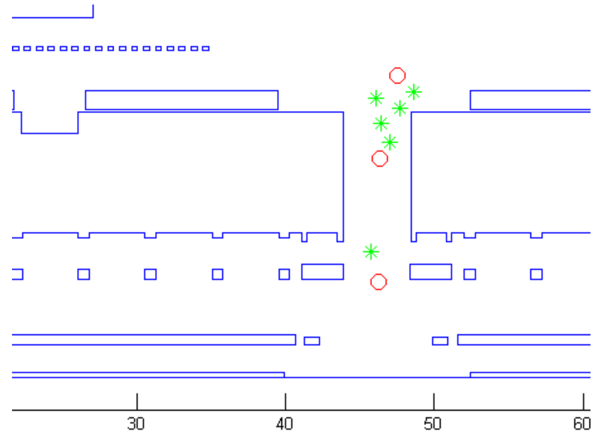

Figure 5. A group of people is being guided by three robots, one people has escaped from the formation, and one of the shepherd robots have rescued him and is accompanying him to the group.

terfaces to communicate between components (schemas). This middleware is enough flexible to allow the programation of low level tasks as obstacle avoidance or to program high level task as this task allocation.

We will present different simulations of a group of robots guiding several people. We have studied which are the different behaviors of robots and how are distributed the tasks depending on people behaviors. In the sequences of Fig. 5-8, we show some instants of time of the different simulations we have developed. Robots are represented with circles, and people with asterisks.

As it has been mentioned several times in this paper, we are trying to study who are distributed the task we can find in a guiding people mission. For that reason, it is necessary to evaluate how different people behaves affect on the task allocation.

We have developed some simulations of guiding a group of 5 or 6 people by two or three robots in a 2D with obstacles, where the group of people has to reach an specific goal. In several occasions the Task Allocation have more than three robots, however, in other occasions, the systems only can use two robots.

In Fig. 5-8, some instant of simulations were presented, in all simulations performed, the systems is able to solve different tasks by different ways, this new allocation have the advantage that though, a task is being solved by one robot correctly, other robots can join and solve the assigned task by a better way.

Finally, Fig. 9, shows robots and people trajectories while the group is being guided, here we can see how people follows leader's trajectory untill arrive to their goal.

On the submited video, we show a guiding task where a person is distracted and lose the group, Then another robot performing surveillance change their task and guide the distracted person to reinstate the group.

6 CONCLUSIONS AND FUTURE WORKS

This paper has been focused on the description of the selfish task allocation model to carry out guided people in urban areas with a set of mobile robots working in a cooperative manner, working with the model DTM the robot can act without the need to be constantly watching the movement of people. We have presented some results in which it has been applied the method described and it has become apparent that the robots can act early enough to guide group of people.

Robots perform their task by a cooperative way, and are able to solve new tasks, for instance, rescuing people who are escaping from the formation, here, the Selfish Task allocation have salved the problems by two different ways: one of the robots who are acting as a shepherd robots avoid people escaping from the crowd formation, or another robot, who is not involved in the guidance mission, rescue this person. Another advance is the proposal of a new uncertainty function that allows the election of the best task to perform.

Future studies that must be consider are the following ones: (i) Obtain the optimal number of robots to act depending on the number of people and the environment for optimize the resolutions tasks, and (ii) Study which will be robot behavior if the number of people who are escaping is bigger than the number of robots that the system disposes.

References

[1] M. Arulampalam, S. Maskell, N. Gordon, T. Clapp, D. Sci, T. Organ, and S. Adelaide. A tutorial on particle filters

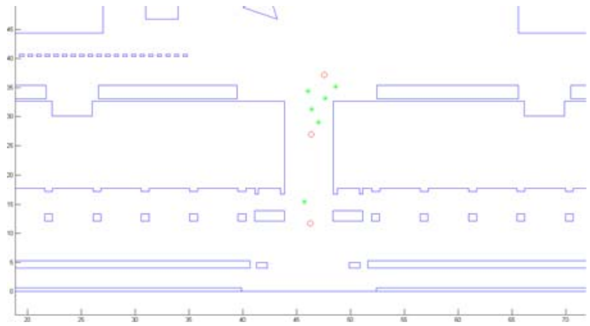

Figure 6. A group of people is being guided by two robots, one people has escaped from the formation, and one robot who was not collaborating in the guiding mission have rescued him and is accompanying to the group.

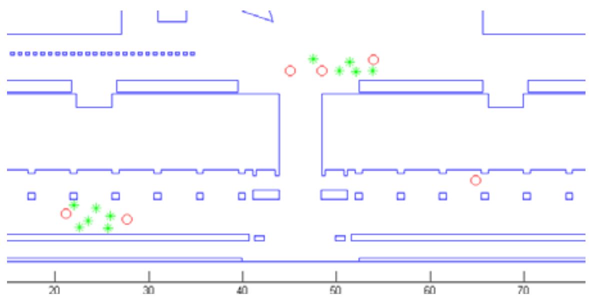

Figure 7. Two different group of people are being guided by groups of robots. The Selfish Task Allocation has distributed the robots between these two independent task, and those tasks are solved properly.

for online nonlinear/non-gaussianbayesian tracking. *Signal Processing, IEEE Transactions on [see also Acoustics, Speech, and Signal Processing, IEEE Transactions on]*, 50(2):174–188, 2002.

[2] S. C. Botelho and R. Alami. M+: a scheme for multi-robot cooperation through negotiated taskallocation and achievement. In *Robotics and Automation, Proceedings of the 1999 IEEE International Conference on*, volume 2, pages 1234–1239, 1999.

[3] C. Breazeal, A. Brooks, J. Gray, G. Hoffman, C. Kidd, H. Lee, J. Lieberman, A. Lockerd, and D. Mulanda. Humanoid robots as cooperative partners for people. *Int.Journal of Humanoid Robots*, 1(2), 2004.

[4] B. Brumitt and A. Stentz. Grammps: A generalized mission planner for multiple mobile robots in unstructured environments. In *IEEE International Conference on Robotics and Automation*, pages 1564–1571. IEEE, 1998.

[5] W. Burgard, A. B. Cremers, D. Fox, D. Hahnel, G. Lakemeyer, D. Schulz, W. Steiner, and S. Thrun. The interactive museum tour-guide robot. *In Proc.of the Fifteenth*

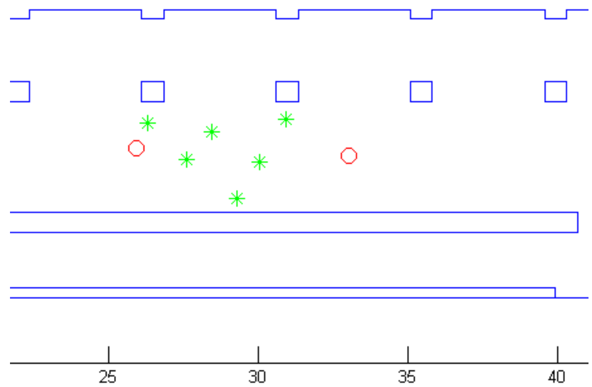

Figure 8. A group of people is being guided by two robots, in this occasion no human escapes from the formation, the task is being done correctly, so the Selfis Task Allocation do not sent any other robot because it is not necessary.

National Conference on Artificial Intelligence, Madison, Wi, 1998.

[6] P. Caloud, W. Choi, J. Latombe, C. Le Pape, and M. Yim. Indoor automation with many mobile robots. In *Proceedings of the IEEE International Workshop on Intelligent Robots and Systems*, pages 67–72, 1990.

[7] J. Casper and R. R. Murphy. Human-robot interactions during the robot-assisted urban search and rescue response at the world trade center. *IEEE transactions on systems, man, and cybernetics.Part B, Cybernetics : a publication of the IEEE Systems, Man, and Cybernetics Society*, 33(3):367–385, 2003. PUBM: Print; JID: 9890044; ppublish.

[8] L. Chaimowicz, M. F. M. Campos, and V. Kumar. Dynamic role assignment for cooperative robots. In *Robotics and Automation, Proceedings of the 2002 IEEE International Conference on*, volume 1, pages 293–298, 2002.

[9] A. Corominas, J. M. Mirats, O. Sandoval, and A. Sanfeliu. Real-time software for mobile robot simulation and experimentation in cooperative environments. In S. C. et al., editor, *Simulation, Modeling, and Programming for Autonomous Robots, Proceedings of the First International Conference on*, Lecture Notes in Computer Science, pages 135–146, Venice, Italy, 2008. Springer-Verlag.

[10] K. Dautenhahn, S. Woods, C. Kaouri, M. L. Walters, K. L. Koay, and I. Werry. What is a robot companion friend, assistant or butler? *Proc. Int. Conf. Intelligent Robots and Systems (IROS)*, 25:1488–1493, 2005.

[11] J. Desai, J. Ostrowski, and V. Kumar. Modeling and control of formations of nonholonomic mobile robots. *Robotics and Automation, IEEE Transactions on*, 17(6):905–908, 2001.

[12] M. Dias. *TraderBots: A New Paradigm for Robust and Efficient MultiRobot Coordination in Dynamic Environments*. PhD thesis, Carnegie Mellon University, 2004.

[13] M. B. Dias and A. Stentz. A market approach to multirobot coordination. Technical Report CMU-RI-TR-01-

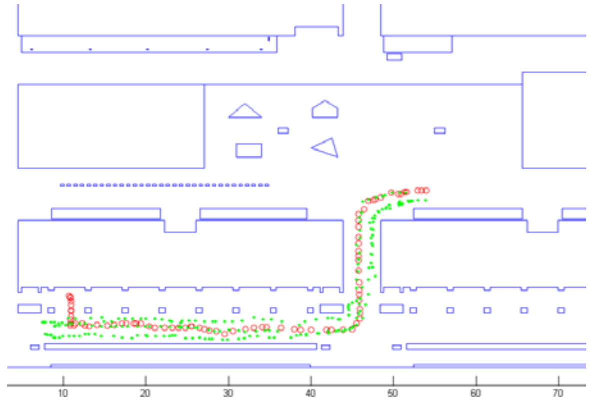

Figure 9. Entire trajectory of people and robots in a guiding people mission.

26, Robotics Institute, Carnegie Mellon University, August 2001.

[14] M. G. E.A., A. O., and S. Yuta. Crowding and guiding groups of humans by teams of mobile robots. *Advanced Robotics and its Social Impacts, 2005.IEEE Workshop on*, pages 91–96, 2005.

[15] A. Garrell, A. Sanfeliu, and F. Moreno-Noguer. Discrete time motion model for guiding people in urban areas using multiple robots. *IEEE/RSJ International Conference on Intelligent Robots and Systems, Saint Louis*, pages 486–491, 2009.

[16] B. P. Gerkey and M. J. Mataric. Sold!: auction methods for multirobot coordination. *Robotics and Automation, IEEE Transactions on*, 18(5):758–768, October 2002.

[17] J. Lien, S. Rodriguez, J. Malric, and N. Amato. Shepherding behaviors with multiple shepherds. *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2005.

[18] L. E. Parker. *Heterogeneous Multi-Robot Cooperation*. PhD thesis, Massachusetts Institute of Technology, February 1994.

[19] L. E. Parker. Alliance: an architecture for fault tolerant multirobot cooperation. *Robotics and Automation, IEEE Transactions on*, 14(2):220–240, 1998.

[20] L. E. Parker. Adaptive heterogeneous multi-robot teams. *Neurocomputing*, 28(1-3):75–92, October 1999.

[21] R.T.Vaughan, N. Sumpter, J. Henderson, A. Frost, and S. Cameron. Experiments in automatic flock control. *Journal Robot and Autonom. Sys.*, pages 191–209, 2000.

[22] A. Sanfeliu and J. Andrade-Cetto. Ubiquitous networking robotics in urban settings. In *Workshop on Network Robot Systems. Toward Intelligent Robotic Systems Integrated with Environments. Proceedings of*, pages 14–18, Beijing, October 2006.

[23] W. M. Shen and B. Salemi. Distributed and dynamic task reallocation in robot organizations. In *Robotics and Automation, Proceedings of the 2002 IEEE International Conference on*, volume 1, May 2002.

[24] R. Smith. The contract net protocol: High-level communication and control in a distributed problem solver. *IEEE Transactions on Computers*, C-29(12), 1980.

[25] P. Stone and M. Veloso. Task decomposition, dynamic role assignment, and low-bandwidth communication for real-time strategic teamwork. *Artificial Intelligence*, 110(2):241–273, 1999.

[26] F. Tang and L. E. Parker. Asymtre: Automated synthesis of multi-robot task solutions through software reconfiguration. In *Robotics and Automation, Proceedings of the 2005 IEEE International Conference on*, pages 1501–1508, April 2005.

[27] F. Tang and L. E. Parker. Coalescent multi-robot teaming through asymtre: a formal analysis. In *Advanced Robotics, Proceedings of the 12th International Conference on*, pages 817–824, July 2005.

[28] Y. Tang and L. E. Parker. Towards schema-based, constructivist robot learning: Validating an evolutionary search algorithm for schema chunking. In *Robotics and Automation, Proceedings of the IEEE Conference on*, 2008.

[29] C. Trevai, Y. Fukazawa, J. Ota, H. Yuasa, T. Arai, and H. Asama. Cooperative exploration of mobile robots using reaction-diffusion equation on a graph. *Robotics and Automation, 2003.Proceedings.ICRA'03.IEEE International Conference on*, 2, 2003.

[30] A. Viguria, I. Maza, and A. Ollero. Set: An algorithm for distributed multirobot task allocation with dynamic negotiation based on task subsets. In *Robotics and Automation, 2007 IEEE International Conference on*, pages 3339–3344, 2007.

[31] B. B. Werger and M. J. Mataric. Broadcast of local eligibility: behavior-based control for strongly cooperative robot teams. In *Autonomous agents, Proceedings of the fourth international conference on*, pages 21–22, New York, USA, 2000. ACM.

[32] D. Zhang, G. Xie, J. Yu, and L. Wang. Adaptive task assignment for multiple mobile robots via swarm intelligence approach. *Robotics and Autonomous Systems*, 55(7):572–588, 2007.