Autonomous Robots 8, 87–97 (2000)

Robot Awareness in Cooperative Mobile Robot Learning

CLAUDE F. TOUZET
Center for Engineering Science Advanced Research, Computer Science and Mathematics Division, Oak Ridge National Laboratory, P.O. 2008, Oak Ridge, TN 37831-6355, USA
touzetc@mars.epm.ornl.gov

Abstract. Most of the straight-forward learning approaches in cooperative robotics imply for each learning robot a state space growth exponential in the number of team members. To remedy the exponentially large state space, we propose to investigate a less demanding cooperation mechanism—i.e., various levels of awareness—instead of communication. We define *awareness* as the perception of other robots locations and actions. We recognize four different levels (or degrees) of awareness which imply different amounts of additional information and therefore have different impacts on the search space size ($\Theta(0)$, $\Theta(1)$, $\Theta(N)$, $o(N)$,[1] where N is the number of robots in the team). There are trivial arguments in favor of avoiding binding the increase of the search space size to the number of team members. We advocate that, by studying the maximum number of neighbor robots in the application context, it is possible to tune the parameters associated with a $\Theta(1)$ increase of the search space size and allow good learning performance. We use the cooperative multi-robot observation of multiple moving targets (CMOMMT) application to illustrate our method. We verify that awareness allows cooperation, that cooperation shows better performance than a purely collective behavior and that learned cooperation shows better results than learned collective behavior.

Keywords: cooperative robotics, cooperative learning, robot awareness, CMOMMT, lazy reinforcement learning

1. Introduction

Cooperative behavior is a subclass of collective behaviors (i.e., any behavior of robots in a system having more than one robot). Cao et al. (1997) in their recently published extended survey of the cooperative mobile robotics field define cooperative behavior as follows: *"Given some task specified by a designer, a multiple-robot system displays cooperative behavior if, due to some underlying mechanism (i.e., the* "mechanism of cooperation"*), there is an increase in the total utility of the system.*" The mechanism of cooperation may lie in the imposition by the designer of a control or communication structure, in aspects of the task specification, in the interaction dynamics of robot behaviors, etc. We dismiss the obvious choice of "communication", preferring to promote "robot awareness"—a less complicated issue—as the necessary basic component for cooperation. Awareness encompasses the perception of other robot's locations and actions.

No previous research has investigated awareness in the context of learning. Related work in robot awareness includes Parker (1995) but that study was restricted to the human-designed policy case. Learning involves the exploration of the search space to gather information about the task, and exploitation of the data, usually through generalization. The main restriction to the use of learning comes from the size of the search space—the larger the search space the more difficult the generalization. Awareness of other robots implies the addition of several dimensions to the search space (compared to an application involving a unique robot or to a pure collective behavior). We recognize four degrees of awareness of other team members and each one impacts the search space size differently. In this paper, we propose a method to select—before starting the learning and in relation with the application—the awareness degree and set its parameter. The cooperative multi-robot observation of multiple moving targets (CMOMMT) application will serve as an illustration.

Since cooperative robot learning raises, at least, all the issues attached to robot learning, we review in the following Section 2 several considerations associated with single robot learning. In Section 3, we describe four different degrees of awareness and how they impact the search space size. The following sections are devoted to the setting of awareness parameters that will allow cooperation without an unmanageable search space size. Experimental results presented in Section 4 describe the effects of a limited range and a bounded arena on the robot's awareness. Section 5 studies the relation between the total number of neighbor robots and the robot policy. Beginning Section 6, we use the CMOMMT as an illustrative application to report on the performance associated with robot awareness. Our first experiment verifies the positive effect of awareness on the performance using human-design policies. In the following Section 7, we plot the relation between the robot awareness range and the performance in CMOMMT. Then, Section 8 presents the results obtained using a lazy reinforcement learning approach. We review related works in Section 9. Finally, we summarize and offer concluding remarks.

2. Cooperative Robot Learning

Cooperative robot learning can be defined as the automatic modification of the robot behaviors to improve the team performance in its environment. At least, cooperative learning presents all the issues associated with individual robot learning. These issues are related to the intrinsic nature of the robots, the complexity of the task to learn and the necessary involvement of generalization.

2.1. Robot's Nature

Robots are by definition artifacts using numerical sensors and actuators to deal with the real world. They are requested to either address today's unsolved symbol grounding problem (Brooks, 1991), or to rely on sub-symbolic processing. As long as the grounding problem of symbols is not solved, symbolic methods cannot be used (at least alone). So, the burden of cooperative learning in robotics falls on the sub-symbolic approaches. Numerical sensors and actuators allow us to define—roughly—a computational measure of the search space size. If d is the number of sensors, p the number of possible sensor readings, and we assume that all sensors share the same p, then the search space size is equal, in a first approximation, to p^d.

2.2. Exploration Technique

The primary goal of learning is to provide—automatically—an increase of the performance of the robot behavior. There are two main sub-symbolic approaches used in robot learning, they differ by the way the exploration is accomplished. Supervised learning lets the human operator do the exploration of the search (or situation) space. The learning algorithm will convert it to an exploration of the space of possible policies. Then, the effective size of the search space has no influence on the learning—as long as the selected examples are representative. The number of learning samples depends on the size of the possible policies space, but not (at least not directly) on the situation space size. Supervised learning implies that the human operator knows how to execute the task given to the robot, or at least knows how to select the relevant examples representative of the task.

Reinforcement learning (Watkins, 1989; Sutton et al., 1998) changes the task description level, only requiring from the human operator a performance measure of the desired behavior (Kaelbling et al., 1996; Dorigo, 1996). In reinforcement learning, exploration is a necessary step. In the absence of bias (discussed in the next paragraph), the exploration process searches the entire situation-action space.

Due to the difficulties associated with building the learning sample base in supervised learning, even for applications involving a unique robot (Heemskerk et al., 1996), we select reinforcement learning as our paradigm for cooperative learning.

2.3. Limited Number of Samples

Even without involving the battery life time, which is restricted in the better case to a few hours, the mechanical nature of the actuators only allows a limited number of actions to be performed during an experiment. To insure the convergence of the learning phase, despite this limited number of available samples, two—non exclusive—different approaches are proposed: generalization and biases. Neural-based reinforcement learning implementations have demonstrated high efficiency in generalization (Lin, 1992; Sehad et al., 1994; Kretchmar et al., 1997). The number

of samples needed to estimate a function of several variables to a given level of accuracy grows exponentially with the number of variables. Therefore, the generalization performance is proportional to the ratio number of samples over the search space size. A huge search space normally limits the performance of the learning and it is common practice to reduce the search space size by using biases (Santos et al., 1998). Numerous biases have been described in the literature and can be ranked using the amount of the search space left for exploration (Touzet, 1998). The most drastic ones reduce so much the size of the situation-action space that a complete, or near complete, exploration becomes possible (Mataric, 1997a). In (Mataric, 1997b) for example, there are efficient foraging policies that take into account the local distribution of the pucks, or the other robot's positions, and which cannot be obtained by the a priori given repertoire of fixed behaviors (safe wandering, dispersion, resting, homing) and the predicate conditions (have puck, at home, near intruder, night time). In Mataric's case, the small size of the search space impedes the development of unforeseen learned solutions, and learning does not apply when a complete modelization or a complete exploration is available. A more limited use of biases—at the cost of the necessary involvement of generalization techniques (Touzet, 1997)—reduces the search space size without jeopardizing the learning.

3. Robot Awareness Degree and the Search Space Size

Specifically associated with cooperative mobile robotics is the need for each robot to take into account the others. Communication, because it implies an emitter, a receiver, a message, etc., is a very complex way to achieve cooperation. It is our opinion that awareness of other team member's positions and actions is more appropriate, in particular in the context of subsymbolic learning. It may not be always feasible to obtain awareness without communication, but this is an independent issue. Parker (1995) distinguishes three approaches of robot awareness from implicit awareness through a teammate's effect on the world (no explicit interaction between the robots), to passive observation of a teammate's actions or goals (result from robots sensing one another), to explicit communication of a teammate's actions or goals.

Such taxonomy is interesting to classify between cooperative robotics applications, but it does not help when it comes to building a cooperative learning application. It is more useful to evaluate robot awareness through its impact on the number of the robot inputs and, therefore, on the search space size. In Fig. 1, we distinguish four degrees of robot awareness of other members of the team. Figure 2 displays the number of dimensions d (i.e., robot's inputs) of the search space size vs. the number of robots for each degree of awareness. Let us remember that the search space size $= p^d$, with p the number of possible sensor readings.

3.1. $\Theta(0)$ *Additional Information*

This case is the lower bound in term of search space size increase (i.e., no increase). The existing situation inputs (n) are sufficient for coding information about the other group members. This is the interaction via environment case. The problem is to extract (before any cooperation) the information relative to the other robots from the input world situation—not an easy task. For example, when Premvuti and Yuta (1996) consider communication for mobile robots, they emphasize the need of

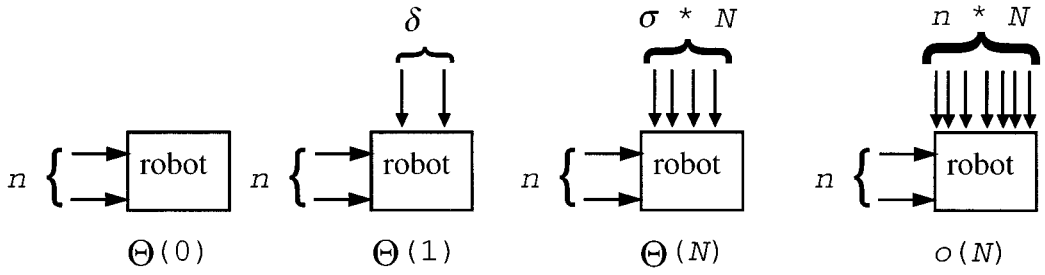

Figure 1. The four degrees of awareness that a robot can exhibit (relative to the other N members of the group) and their impact on the number of inputs of the search space size of the individual robot. n is the number of sensors used to perceive the world situation. δ is a fixed set of additional inputs to represent the knowledge about all the other members of the group ($\delta < N$). σ is a set of additional inputs used for each other member ($N \leq \sigma < n * N$).

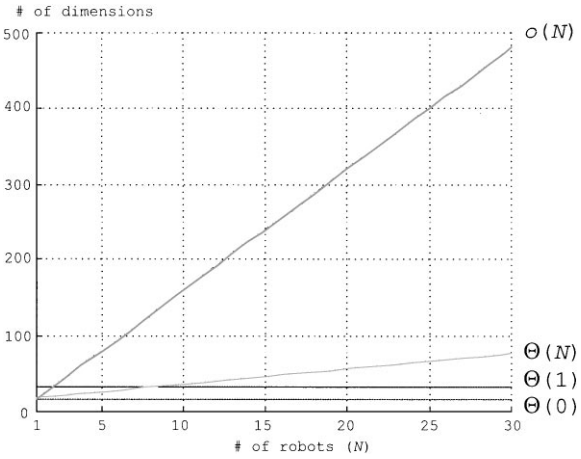

Figure 2. Number of dimensions of the search space size vs. number of robots (N) for each of the 4 degrees of awareness ($\Theta(0)$, $o(N)$, $\Theta(N)$, $\Theta(1)$). The parameter values (cf. Fig. 1) are $n = 16$, $\delta = 16$, $\sigma = 2$ and $N = [1, 30]$.

so-called implicit communication during cooperation, but not surprisingly, the authors conclude suggesting that, with the current technology, "implicit communication" should be done through the help of communication network of the multiple robot system—i.e., explicit communication.

3.2. $o(N)$ Additional Information

This case is the upper bound in term of search space size increase (i.e., maximum increase). The objective is to get as much information as possible, for example by sharing the n inputs of the other N robots. The result is comparable as having duplicated sets of central supervisors, one for each individual. The search space size is the combination of the individual search spaces ($n * N$).

3.3. $\Theta(N)$ Additional Information

A less demanding case compared to $o(N)$ is to use a limited awareness. For example, our application field being mobile robotics, we can choose to use orientation and distance to another robot as the pertinent information. The number of inputs associated here ($\sigma = 2$) is much smaller then the n previously requested. The search space size is $\sigma * N$. Each robot of the team is taken into account.

3.4. $\Theta(1)$ Additional Information

With the previous degree of awareness, the number of robots has a direct, and dramatic, influence on the search space size. We would like to be able to provide awareness independently of the number N of robots, for example by using a fixed set of additional inputs (δ) to represent the knowledge about the other members of the group (how to obtain such knowledge is not relevant in this paper). The limitation (related to a fixed amount of additional knowledge space) is that the individual labeling of each group member is impossible as soon as the number of robots surpasses the number of added inputs. This will not be a problem if we can verify that cooperation is nevertheless achieved using this awareness level. The question to answer is "What should be the value of δ so that there is no difference in awareness quality with degree $\Theta(N)$ (with $\delta < \sigma * N < n * N$)?"

4. Number of Neighbor Robots (Non-Cooperative Policy)

Real applications imply a limited range of the robot awareness, which means that certainly only a subset of all the team members can be sensed at a given time by a member of the team. We will use that observation to compute the value of the parameter δ. A bounded arena, by limiting the spreading of the robots, has certainly a counter-effect. Figure 3 shows the bounded arena and the robots equipped with a 360° field of view of limited range.

The mean value of the number of robots sensed by any member of the group can be easily computed.

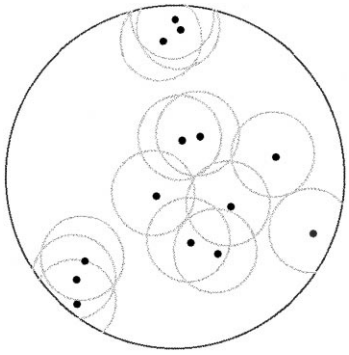

Figure 3. Bounded arena with 14 robots moving randomly. The radius of the arena is 5, the radius of the sensory perception range of the robots is 1. At the top of the figure, 3 robots sense each other.

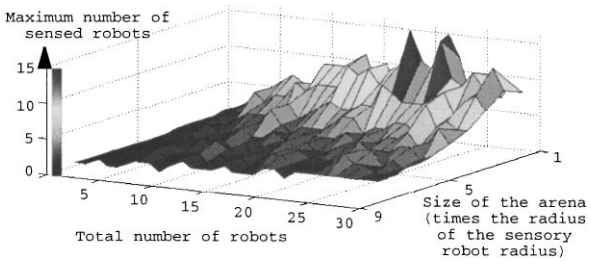

Figure 4. Maximum number of robots perceived by a robot vs. the number of robots ($N = [1, 30]$) and vs. the radius of the bounded arena $[1, 9]$. The behavior of the robots is random walk. The sensory perception radius for each robot is 1. The results displayed are the maximum of 5 different experiments, each one of 600 iterations. Remember that for a robot at the border of the arena the sensory perception area is not maximum (part of it lays outside the arena), which explains why, for an arena size of 1, the number of perceived robots is not equal to the team size.

However, cooperation and, therefore robot awareness, is particularly needed when there are a lot of neighbor robots. So, instead of the mean value of the number of robots, we prefer to study the maximal number of sensed robots by a team member. Over an infinite period of time, the maximal number of sensed robots would be the number of team members (less one). However, typical application time length is much shorter (a time period of 600 moves (per robot) has been selected). Figure 4 shows the maximum number of robots perceived by a robot in respect to the size of the arena and the total number of robots. The robot policy is random walk, the robots are initially spread randomly (uniform distribution) over the entire arena. These experimental conditions will also be the initial conditions of the learning (exploration phase). A value of $\delta = 16$ seems appropriate. We will verify in the next sections that it is appropriate under conditions closer to the selected application (CMOMMT).

5. Number of Neighbor Robots (Cooperative Policy)

In the previous sections (4 & 5), we assume that the robot policy is random walk. A number of policies would certainly allow a better spatial distribution of the robots, but will they reduce de facto the number of robots within sensory range? In the CMOMMT (Parker, 1997) application, a team of robots with 360° field of view sensors of limited range has to maximize the observation time of a set of targets moving randomly (5% probability to change direction, maximum speed less than the maximum robot speed), in a bounded arena. We say that a robot is monitoring a target when the target is within that robot's observation sensory field of view. The objective is to maximize the collective time during which targets are being monitored by at least one robot. The radius of the sensory robot range is less than the size of the arena, implicating that robots have to move to maintain observational contact. In this context, A-CMOMMT (Parker, 1997) is certainly the most effective human-designed robot policy. It combines low and high level control algorithms. Local control of a robot team member is based upon a summation of force vectors, which are attractive for nearby targets and repulsive for nearby robots. High-level reasoning control involves the computation of a probability that no other robot is already monitoring the target and a probability that a target exists, modeled as a decay function based upon when the target was most recently seen, and by whom.

Results displays in Fig. 5 show that for a large number of robots (5 to 30), as long as the radius of the arena is not ridiculously small (e.g., 30 robots in an arena of size 1), the number neighbor robots does not vary very much (around 7). This suggests that cooperation policy (here A-CMOMMT) has an impact on the distribution of the robots, and therefore on the maximum number of sensed robots.

However, it must be emphasized that learning—in particular, in its early exploration stage if starting in *tabula rasa* condition—will be much closer to non-cooperative policy than cooperative ones. Therefore, a selection of δ based on a non-cooperative behavior is coherent. The number of available targets has little effect on the results since the built-in awareness in

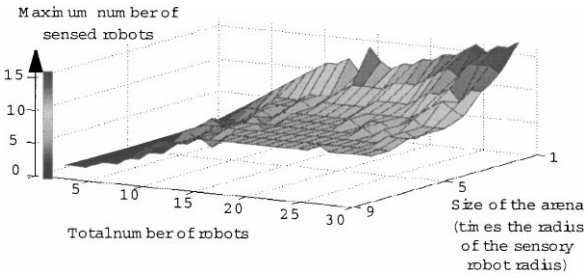

Figure 5. Maximum number of robots perceived by each robot vs. the number of robots ($N = [1, 30]$) and vs. the radius of the bounded arena $[1, 9]$. Each robot uses an A-CMOMMT policy. The sensory perception radius for each robot is 1. The results displayed are the maximum of 5 different experiments, each one of 600 iterations. Compared to Fig. 4, we see that the highest values are almost identical, which confirm our previous conclusion.

A-CMOMMT allows only the nearest robot to follow a given target. On the contrary, the behavior of the targets (e.g., targets avoiding robots) may have a huge impact on the number of robots within sensory range—but it will have no influence during the early stage (exploration) of the learning.

6. Collective vs. Cooperative Policies

An important issue is to verify that robot's awareness has an impact on the team performance when accomplishing its task. We use CMOMMT as a benchmark application. In Fig. 6, we plot the performances of the team with *no* robot awareness vs. the size of the arena and vs. the number of robots. The performance is computed as the percentage of observed targets (by the group). There are 15 targets. Each robot is equipped with a behavior that places it at the geographical center of the sensed targets; it does not take into account the other robot positions.

The experimental results point out several surprising things: the performance variation is, on the average, small; the influence of the number of robots is only slightly perceptible; the initial drop of performance for very small arena sizes (1–2) is huge and very large arena sizes allow better performances than smaller ones.

The logical explanation consistent with these results is that, having no knowledge about other robot positions, a robot often chooses to track an already tracked target. Despite the fact that initialization spreads robots and targets uniformly over the entire arena surface, during the 600-iterations experiment there are lots of

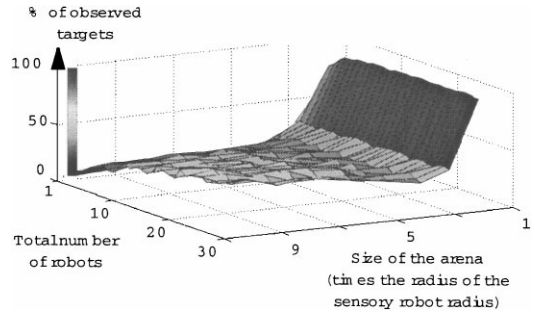

Figure 6. Percentage of observed targets (by the group) vs. the number of robots ($N = [1, 30]$) and vs. the radius of the bounded arena $[1, 9]$. Each robot uses a non-cooperative policy that tries to place it at the geographical center of the sensed targets. There are 15 randomly moving targets. The sensory perception radius for each robot is 1. The results displayed are the mean of 5 different experiments, each one of 600 iterations.

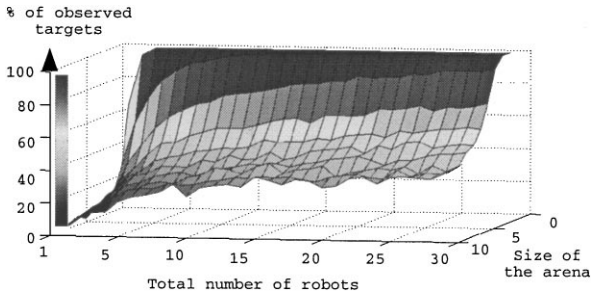

Figure 7. Percentage of observed targets (by the group) vs. the number of robots ($N = [1, 30]$) and vs. the radius of the bounded arena $[1, 9]$. Each robot uses the A-CMOMMT policy, and therefore is aware of the position of the other robots in its sensor field of range. There are 15 randomly moving targets. The sensory perception radius for each robot is 1. The results displayed are the mean of 5 different experiments, each one of 600 iterations.

opportunities for a robot to become useless in following already tracked targets. However, this explanation does not work for larger arena results. In fact, in this case, there is less chance for a given robot to encounter already tracked targets. The initial impressive performance for an arena size of 1 comes from the fact that, since the robot perception radius is also 1, there is only a very small influence of the robot policy on performance.

Using A-CMOMMT policy for the robots (i.e., adding robot awareness), we obtain the results displayed in Fig. 7 (same conditions as Fig. 6). Here, the performance is monotonic in respect to the size of the arena and the number of robots. A maximum value of 100% is easily reached for small arenas, and an increased size for the arena surface implies a continuous decrease for the percentage of targets under observation. There are only 15 targets in the arena, so the advantage of additional robots after 8–10 is limited.

Figure 8 plots the increase in performance associated with the use of robot awareness (A-CMOMMT policy, Fig. 7) vs. the purely collective behavior (cf. Fig. 6). The difference in performance can reach 60%. The impact of robot awareness is particularly notable with small arena sizes (2–6), pointing out the advantage of being able to maintain a minimal distance between team members and being able to select untracked targets. The effect of the number of robots is logically positive for small arena size (2–6), and becomes null for large arenas, where robots (and targets) have so much space available that they do not come close to each other anymore. The limited robot awareness range implies that after a given size of the arena (8–9), robot

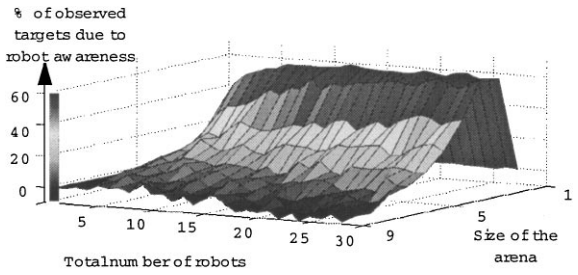

Figure 8. Percentage of observed targets due to robot awareness vs. the number of robots ($N = [1, 30]$) and vs. the radius of the bounded arena [1, 9]. Each robot uses the A-CMOMMT policy, and therefore is aware of the position of the other robots. There are 15 randomly moving targets. The sensory perception radius for each robot is 1. This graph shows the difference between Fig. 6 results and Fig. 7 results (i.e., cooperation vs. collective policies).

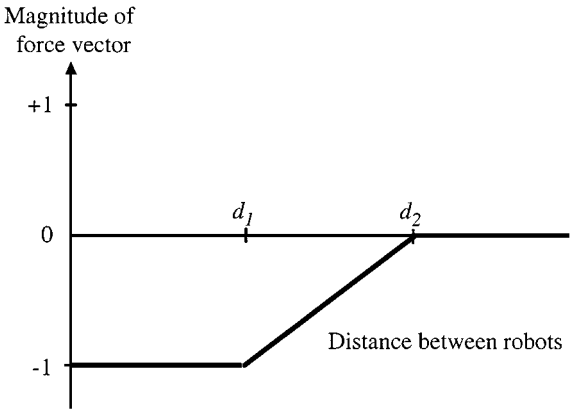

Figure 9. Function defining the magnitude of the repulsive force vector to nearby robots.

awareness is of no more use. The effect of large robot numbers is to slow the disappearance of the usefulness of robot awareness (particularly visible for arena sizes between (2–7)).

It must be note that despite the impression that multi-robots systems performs well only for small values of arena size, the performance of mobile robots is way above that of stationary ones (even well placed). Let us compute the ratio arena under observation versus arena surface for a group of stationary non overlapping robots. We take the case of a group of 30 robots, an arena of radius 10 and 15 targets (cf. Fig. 7). The ratio is equal to 30% of the surface under observation, which account for an average number of 4.5 targets under observation, to compare with the number of 7.5 reported on Fig. 7.

7. Influence of the Robot Awareness Range on the Performance

The previous section has reported the large impact on the performance of the robot awareness. Certainly, the larger the robot awareness range, the better the performance is. However, a large robot awareness range implies a large number of neighbor robots to take into account. It is not desirable to allow a too large number of sensed robots, because of its effect on the dimensionality of the search space. In fact, it would be interesting to be able to reduce as much as possible the robot awareness range.

In this section, we study the influence of the robot awareness range on the performance. In the case of A-CMOMMT policy, the repulsive force between the robots is defined in Fig. 9. If the robots are too close together ($<d_1$), they repel strongly. If the robots are far enough apart ($>d_2$), they have no effect upon each other in terms of the force vector calculations. The repulsive force magnitude scales linearly between these cases ($[d_1, d_2]$).

The increase in the surface awareness area is proportional to the square of the robot awareness range (e.g., if the range is multiplied by 2, then the surface is multiplied by 4). Figure 10 displays the additional percentage of observed targets due to a robot awareness range multiplied by 2 (from a range of 0.5 to a range of 1). The robot policies are A-CMOMMT and there are 15 randomly moving targets. We see that a larger robot awareness range automatically implies a better performance. Reducing the dimensionality of the search

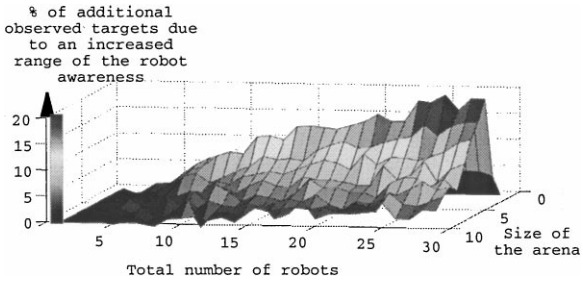

Figure 10. Percentage of additional observed targets due to a robot awareness range multiplied by 2 (from a range of 0.5 to a range of 1) vs. the number of robots ($N = [1, 30]$) and vs. the radius of the bounded arena [1, 9]. The robot policies are A-CMOMMT and there are 15 randomly moving targets. The sensory perception radius for each robot is 1. The results displayed are the mean of 5 different experiments, each one of 600 iterations.

space—the number of neighbor robots—by a decrease of the robot awareness range has a direct negative impact on the performance and should be avoided. A compromise has to be found between a too large number of neighbors and a too small number of neighbors.

8. Cooperative vs. Collective Learning

Until now, we have studied the impact of robot awareness upon learning (in fact, the search space size) without the help of any learning experiment. Our case study sets the parameter δ (so that we are in a $\Theta(1)$ awareness degree) to 16. Therefore, we will use 16 additional inputs to represent the information about neighbor robots in CMOMMT learning experiments. Each robot situation is a vector of $2*16$ components.[2] The first 16 components code the position and orientation of the targets. It simulates a ring of 16 sensors uniformly distributed around the robot body. Each sensor measures the distance to the nearest target. The sensor position around the body gives the orientation. The second ring of 16 components codes in the same manner the position and orientation of neighbor robots (how to distinguish between targets and robots is not relevant here, but certainly the sonar values have to be completed with other information). The maximum range allowing a target or a robot to be seen is 1. The actions of each robot are rotation and forward move distance. With the objective of reducing the number of actual moves during the behavior synthesis—and therefore the time required by an experiment—we use a lazy learning approach (Aha, 1997).

8.1. Lazy Learning

In a lazy learning approach, the computation of the inputs is delayed until the necessity arises. In a first phase, lazy learning samples the situation-action space and stores the succession of events in memory. In a second phase, lazy learning probes the associative memory for the best move. The sampling process stores the successive situation-action pairs generated by a random action selection policy, whereas the questioning of the memory involves complicated computations: clustering, pattern matching, etc. Using lazy learning and reinforcement function probing in the associative memory (Sheppard et al., 1997), the exploration phase can be done only once, stored and used later by all future experiments. This way, an experiment only requires a test phase: a measure of the performance of the learning. The learning phase occurs during the probing of the memory and involves lots of computations. However, the computation time requirements are negligible compared to the robot mechanical time requirements. This way, an experiment in cooperative robotics is even shorter (effective time) than an experiment involving just one robot and eager learning. It must be emphasized that, because it is independent of the nature of the desired behavior, in lazy learning the initial exploration phase is unique.

Sheppard et al. (1997) propose to probe the memory with the reinforcement function. Their objective is to provide a method for predicting the rewards for some state-action pairs without explicitly generating them. They call their algorithm lazy Q-learning. For the current real world situation, a situation matcher locates all the states in the memory that are within a given distance. If the situation matcher has failed to find any nearby situations, the action comparator selects an action at random. Otherwise, the action comparator examines the expected rewards associated with each of these situations and selects the action with the highest expected reward. This action is then executed, resulting in a new situation. There is a fixed probability (0.3) of generating a random action regardless of the outcome of the situation matcher. New situation-action pairs are added to the memory, along with a Q-value computed in the classical way. Among similar situation-action pairs in the memory, an update of the stored Q-values is made. There is a limit to the genericness of this lazy memory because the Q-values associated with the situation-action pairs only apply for a particular behavior. With the desire of reducing as much as possible the learning time and also of preserving the genericness of the lazy memory, we modified the algorithm in the following way: the situation matcher always proposes the set of nearest situations—no maximum distance is involved—and there is no random selection of actions by the action comparator. Also, the Q-values are not stored with the situation-action pairs, but are computed dynamically as the need arises.

The key to successful application of the lazy Q-learning algorithm is the identification of similar situations. We use a measure of similarity of the following form:

$$similarity(a, b) = \sum_{i}^{p} (|s_a(i) - s_b(i)|) \qquad (1)$$

where s_a and s_b are two situations and p is the number of components of the situation. The smaller the value measured, the greater is the similarity.

8.2. Cooperative Lazy Learning

Cooperative reinforcement learning requires a method to distribute the reinforcement values among the group members. We, and others from the multi-agent community in particular, are pursuing our research efforts in this direction, but our results have not yet reached the quality of the human-defined A-CMOMMT. They will nevertheless allow us to demonstrate the impact of robot awareness in CMOMMT applications. Figure 11 shows the increase of performance associated with robot awareness vs. a purely collective behavior.[3] Each robot behavior (cooperative or collective) is learned through lazy reinforcement learning. The dimensionality of the search space is 32 (targets + robots) for cooperative behavior, and only 16 (targets) for collective behavior. The lazy memory is obtained through an initial exploration (length 120 iterations) involving 10 targets and 5 robots, the policies for targets and robots were random action selection. The reinforcement function we use is the following: $+1$ if one (or more) targets have been acquired compared to the previous situation, -1 if one (or more) targets have been lost, or 0 otherwise. Each iteration the probabilities of direction change for a target was 5%, where it was 100% for a robot. The total number of situation-action pairs in the associative memory is $600(=120 * 5)$.

Compared to Fig. 8, we see that the impact of robot awareness is less noticeable in our learning experiment:

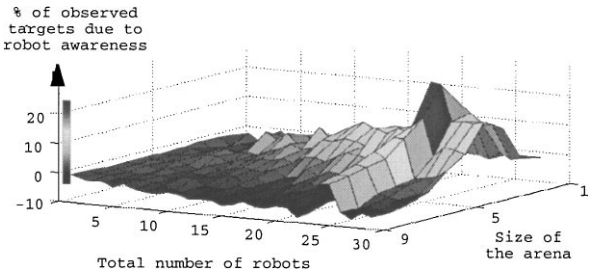

Figure 11. Percentage of observed targets due to robot awareness (cooperative-collective) vs. the number of robots ($N = [1, 30]$) and vs. the radius of the bounded arena $[1, 9]$. Each robot learns its behavior using lazy reinforcement learning. There are 15 randomly moving targets. The sensory perception radius for each robot is 1. This graph is obtained in similar conditions as those of Fig. 8, except for the behaviors, which are learned here.

a maximal improvement of 25% (compared to 60%). However the shape of both surfaces are similar: the preferred arena size is between 2 and 6, and the greater the number of robots, the more important the impact on the performance. The counter-effect of very large arenas is easily spotted, in particular for small numbers of robots. In this case, awareness seems to slightly hurt the performance. This is due to the fact that the collective behavior is learn with a smaller search space size ($\Theta(0)$) compared to ($\Theta(1)$) for the cooperative behavior. Since there is very little, or no effect of cooperation, the ratio of the search space size vs. the number of learning sample is smaller for the collective behavior learning case (allowing better learning performance).

9. Related Works

Ono and Fukumoto (1997) are interested in reducing the search space size so as to allow multi-agent reinforcement learning. The main idea underlying their approach is that each agent's learning component is decomposed into independent modules, each focusing on one agent. The learning results of these components are combined by a mediator using a simple heuristic procedure (the greatest mass merging strategy). Their approach is based on a reduction of the search space size by a decomposition into multiple sub-goals, initially proposed by Whitehead (1993) for one agent. They illustrated the use of their modular architecture with a modified version of the pursuit problem: in a 20×20 toroidal grid world, a single prey and four hunter agents. The behavior of the prey is random walk. A hunter has a limited field of view and can differentiate between prey and hunters. For each other agent, a module is used that takes into account (only) the relative position of that agent and the prey: this is an awareness associated with a $\Theta(N)$ increase in the number of dimensions. The field of view range (i.e., robot awareness range) is small enough to drastically reduce the search space size, even using such awareness degree (cf. Section 8).

Kube and Zhang (1994) simulations of a collective box-pushing behavior involve robots equipped with a goal sensor, an obstacle sensor and a robot sensor. No learning is involved—the robot policies are user-defined. Each behavior is implemented as a "Braitenberg vehicle" (Braitenberg, 1984). The arbitration between behaviors uses fixed priority assignment in a subsumption approach. A second approach tested

for behavior arbitration is to train an adaptive logic network through a supervised learning procedure. The authors do not give details about the way the robot sensor intervenes; therefore, it is impossible to determine the degree of awareness. Moreover, the extremely small number of actions available to insure cooperation (in *follow*, the robot sensor is used to direct the robot to the nearest sensed neighbor; in *slow*, the robot reduces its velocity whenever neighbor robots are detected) tends to deny, in our opinion, the term "cooperation" to this work.

Balch and Arkin (1994) in their desire to create a design methodology for multiagent reactive robotic systems have been interested in choosing correctly the number of agents and the communication mechanisms. They define 4 levels of communication: *no communication* where the robots are able to discriminate between robot, attractors and obstacles, *state communication* where robots are able to detect the internal state of other robots, *goal communication* where the sender must deliberately send or broadcast the information and *implicit communication* which corresponds to communication through the environment (no act of deliberate transmission). It must be pointed out that these authors are not interested in learning, therefore they did not address the issue related to the search space size. Interestingly, their results demonstrate that, at least for the three experimented tasks of forage, consume and graze, cooperation emerges with the "no communication" paradigm. The maximum improvements are: forage task, goal vs. no communication (19%); consume task, state vs. no communication (10%) and graze task, state or goal vs. no communication (1%). Their conclusion is that for some tasks, higher levels of communication can slightly improve performance, but for other inter-agent communication is apparently unnecessary. This last remark agrees with our selection of "robot awareness"—instead of "communication"—as the necessary basic component for cooperation.

10. Conclusion

Cooperative behavior definition points out the necessity for a mechanism of cooperation that we translate in robot awareness of other team members. The search space size is of tremendous importance for the learning ability (the smaller the better), therefore we have presented the different degree of awareness in respect to their influences on the search space size (no awareness: $\Theta(0)$, restricted awareness: $\Theta(1)$, awareness of all: $\Theta(N)$ and complete communication: $o(N)$). We have presented a method to elect a $\Theta(1)$ awareness using the fact that the sensors have limited range. The careful study of the *maximum* number of neighbor robots (instead of the total number of robots) allows to set the parameter δ. The maximum number of neighbor robots is dependent on the arena size, the awareness range, and also on the robot policies. The cooperative multi-robot observation of multiple moving targets (CMOMMT) domain is used as an illustrative application—but our method is generic and can be applied to many applications. It consists in studying the maximum number of neighbor robots in the beginning of the learning phase (exploration) so as to be able to determine the appropriate value of δ. Environmental conditions, like the size of the arena or the range of the sensors, must also to be taken into consideration.

A lazy reinforcement learning approach showed better performance for cooperation than collective behavior and compared well with the best-known human-designed policy (A-CMOMMT). The increase of performance is up to 25% using lazy reinforcement learning and 60% using human-designed policy.

The experimental confirmations were obtained in simulation, but it is our opinion that this does not affect the legitimacy of awareness for cooperative robot learning, nor does it affect the validity of the method.

Acknowledgments

The author thanks Lynne E. Parker for her generous support which includes detailed, insightful and constructive comments, and the anonymous reviewers for their contributions.

This research is funded in part by the Engineering Research Program of the Office of Basic Energy Sciences, US Department of Energy, under contract No. DE-AC05-96OR22464 with Lockheed Martin Energy Research Corporation.

Notes

1. Explanations on this standard asymptotic notation can be found in Cormen T., Leiserson and C. Rivest R., Introduction to Algorithms, MIT Press, 1990.
2. CESAR facilities provide 4 Nomad 200 mobile robots for cooperative robotic experiments. The Nomad 200 is equipped, among other sensory modalities, with a ring of 16 infra-red sensors and another ring of 16 sonar sensors.
3. There are several issues of importance for lazy reinforcement learning performance. The first is the quality of the reinforcement function, the second is the quality of the sampling (size,

representativity), and the third is the quality of the probing process (generalization). The state-of-the-art in these domains only allows us to assume that non-optimal values and criterion were used in the experiments reported here. However, it is our opinion that this does not alter the illustrative quality of Fig. 11.

References

Aha, D. (Ed.) 1997. *Lazy Learning*, Kluwer Academic Publishers.

Balch, T. and Arkin, R. 1994. Communication in reactive multiagent robotic systems. *Autonomous Robots*, 1:27–52.

Braitenberg, V. 1984. *Vehicles: Experiments in Synthetic Psychology*, MIT Press.

Brooks, R. 1991. Intelligence without reason. In *IJCAI'91*, Sydney.

Cao, Y.U., Fukuaga, A., and Kahng, A. 1997. Cooperative mobile robotics: Antecedent and directions. *Autonomous Robots*, 4:7–27.

Dorigo, M. (Guest Editor) 1996. Introduction to the special issue on learning autonomous robots. *IEEE Trans. on Systems, Man and Cybernetics—Part B*, 26(3):361–364.

Heemskerk, J. and Sharkey, N. 1996. Learning subsumptions for an autonomous robot. In *IEE Seminar on Self-Learning Robot*, Digest No: 96/026, Savoy Place, London, 12, England.

Kaelbling, L., Littman, M., and Moore, A. 1996. Reinforcement learning: A survey. *Journal of Artificial Intelligence Research*, 4:237–285.

Kretchmar, R.M. and Anderson, C.W. 1997. Comparison of CMACs and radial basis functions for local function approximators in reinforcement learning. In *Proc. of ICNN'97*, Houston, Texas, USA.

Kube, R. and Zhang, H. 1994. Collective robotics: From social insects to robots. *Adaptive Behavior*, 2:189–218.

Lin, L.J. 1992. Self-improving reactive agents based on reinforcement learning, planning and teaching. *Machine Learning*, 8:293–321.

Mataric, M.J. 1997a. Reinforcement learning in multi-robot domain. *Autonomous Robots*, 4:73–83.

Mataric, M.J. 1997b. Learning social behavior. *Robotics and Autonomous Systems*, 20:191–204.

Ono, N. and Fukumoto, K. 1997. A modular approach to multi-agent reinforcement learning. In *Distributed Artificial Intelligence Meets Machine Learning: Learning in Multi-Agents Environments*, G. Weiss (Ed.), Springer-Verlag, Lecture Notes in Artificial Intelligence, Vol. 1221, pp. 25–39.

Parker, L.E. 1995. The effect of action recognition and robot awareness in cooperative robotic teams. In *Proc. IROS 95*, Pittsburgh, PA.

Parker, L.E. 1997. Cooperative motion control for multi-target observation. *Proc. IROS 97*, Grenoble, France.

Premvuti, S. and Yuta, S. 1996. Consideration on conceptual design of inter robot communications network for mobile robot system. In *Distributed Autonomous Robotic Systems 2 (DARS 96)*, H. Asama, T. Fukuda, T. Arai, and I. Endo (Eds.), Springer-Verlag.

Santos, J.M. and Touzet, C. 1999. Exploration tuned reinforcement function. *Neurocomputing*, 28(1–3):93–105.

Sehad, S. and Touzet, C. 1994. Reinforcement learning and neural reinforcement learning. In *Proc. of ESANN 94*, Brussels.

Sheppard, J.W. and Salzberg, S.L. 1997. A teaching strategy for memory-based control. In *Lazy Learning*, D. Aha (Ed.), Kluwer Academic Publishers, pp. 343–370.

Sutton, R. and Barto, A. 1998. *Reinforcement Learning*, MIT Press.

Touzet, C. 1997. Neural reinforcement learning for behaviour synthesis, Special issue on Learning Robot: The New Wave, N. Sharkey (guest Ed.), *Robotics and Autonomous Systems*, 22(3–4):251–281.

Touzet, C. 1998. Bias incorporation in robot learning, submitted for publication.

Watkins, C.J.C.H. 1989. Learning from delayed rewards, Ph.D. Thesis, King's College, Cambridge, England.

Whithehead, S., Karlsson, J., and Tenenberg, J. 1993. Learning multiple goal behavior via task decomposition and dynamic policy merging. In *Robot Learning*, J. Connell and S. Mahadevan (Eds.), Kluwer Academic Publishers.

Claude F. Touzet is a scientist at the Center for Engineering Science Advanced Research at Oak Ridge National Laboratory (Tennessee). He received his M.Sc. degree in Behavioral Neurosciences in 1985, his Ph.D. degree in Computer Science in 1990 from the University of Montpellier (France) on the subject of sequential neural networks, and his "Habilitation a Diriger des Recherches" in 1998 from University of Marseilles (France). He was a researcher associated with the LERI-EERIE from 1987 to 1994, and an Associate Professor at the University of Marseilles from 1994 to 1997, where he conducted is research at the DIAM-IUSPIM. His research interests are in cooperative robotics, autonomous robotics, reinforcement learning and artificial neural networks. He participates in several projects involving artificial neural networks and robotics with companies or agencies like DARPA (Urban Robot), Renault, Usinor-Sacilor, French Ministry of Defense, K-Team SA, etc.