ELSEVIER

Robotics and Autonomous Systems 20 (1997) 191–204

Robotics and Autonomous Systems

Learning social behavior

Maja J. Matarić [1]

Volen Center for Complex Systems, Computer Science Department, Brandeis University, Waltham, MA 02254, USA

Abstract

This paper discusses the challenges of learning to behave *socially* in the dynamic, noisy, situated and embodied mobile multi-robot domain. Using the methodology for synthesizing *basis behaviors* as a substrate for generating a large repertoire of higher-level group interactions, in this paper we describe how, given the substrate, greedy agents can learn social rules that benefit the group as a whole. We describe three sources of reinforcement and show their effectiveness in learning non-greedy social rules. We then demonstrate the learning approach on a group of four mobile robots learning to yield and share information in a foraging task.

Keywords: Social learning; Group behavior; Reinforcement learning; Social rules; Credit assignment

1. Introduction

Our work focuses on synthesizing complex group behaviors from simple social interactions between individuals [30,32]. We introduced a methodology for selecting and designing a set of *basis behaviors* [2] that serves as a substrate for a large repertoire of higher-level interactions through the application of two general combination operators that allow for overlapping and switching behaviors. Basis behaviors, a direct extension of the behavior-based approach to control, are an effective representation level not only for hard-wired multi-agent control but also for learning. Our previous work, described in [31], has demonstrated how an adaptation of reinforcement learning can be applied to basis behaviors in order to have a group of mobile robots learn a complex foraging behavior in a group.

Although our approach was effective (the group of robots learned to forage within 15 min), the speed of learning declined with increased group sizes, as a result of interference between the agents. In this paper we describe how such interference can be minimized through the use of *social rules*. We discuss the challenges of developing social and altruistic behavior in systems of individually greedy agents facing the group credit assignment problem. To make social learning possible, we postulate three types of social reinforcement, and test their effectiveness in the foraging domain. We demonstrate how a group of robots, initially equipped with a strategy for foraging, can learn the following social behaviors: *yielding, proceeding, communicating*, and *listening*, which serve to effectively minimize interference and maximize the effectiveness of the group.

The rest of this paper is organized as follows. Section 2 overviews some representative related work. Section 3 discusses interference and conflict, the key motivations for social rules, and proposes the concept

[1] E-mail: maja@cs.brandeis.edu.

[2] Alternatively called *basic behaviors*.

PII S0921-8890(96)00068-1

of prototypical states of social interactions that make learning social rules tractable. Section 4 discusses social learning and introduces and motivates the three forms of social reinforcement. Section 5 describes the experimental environment and the robot testbed. Section 6 gives the details of the learning task, the learning algorithm, the implementation of social reinforcement, and the experimental design used for testing. Section 7 presents the results and Section 8 discusses them. Finally, Section 9 describes continuing work and concludes the paper.

2. Related work

While there are many examples of applying learning techniques to simulated mobile robots, there have been comparatively few demonstrations of physical mobile robots learning in real time. This section reviews the work on learning physical mobile robots focusing on reinforcement learning (RL) approaches, and briefly overviews some related learning work on simulated agents.

Kaelbling [19] used a simple mobile robot to validate several RL algorithms based on immediate and delayed reinforcement applied to learning obstacle avoidance. Maes and Brooks [26] applied a statistical RL technique using immediate reward and punishment in order to learn behavior selection for walking on a six-legged robot. The approach was appropriate given the reduced size of the learning space and the available immediate and accurate reinforcement derived from a contact sensor on the belly of the robot, and a wheel for estimating walking progress. More delayed reinforcement was used by Mahadevan and Connell [27] in a box-pushing task implemented on a mobile robot, in which subgoals were introduced to provide more immediate reward. Mahadevan and Conell [28] experimented with Q-learning using monolithic and partitioned goal functions for learning box-pushing, and found subgoals necessary. Lin [21] used RL on a simulated robot by breaking the navigation task into three behaviors in a similar fashion. The work was successfully transferred on a Hero platform. Asada et al. [1] demonstrated coordination of behaviors learned using vision-based reinforcement on a soccer-playing mobile robot shooting at a goal. Tan [47] explored RL in a situated multi-agent domain utilizing communication to share learned information. Lin [22] studied RL in a group of simulated agents.

Chapman and Kaelbing [6] addressed learning from delayed reinforcement in the video game domain. They and Mahadevan and Connell [27] demonstrated complementary approaches for input generalization. Chapman and Kaelbling [6] started with a single most general state and iteratively split it, while Mahadevan and Connell [27] started with a fully differentiated specific set of states, and consolidated them based on similarity statistics accumulated over time.

Aside from traditional unsupervised RL methods described above, other techniques have also been explored. Pomerleau [41] used a supervised connectionist learning approach to train steering control in an autonomous vehicle based on generalizing visual snapshots of the road ahead. Thrun and Mitchell [48] demonstrated a connectionist approach to learning visual features with a camera mounted on a mobile robot. The features are not assigned by the designer but are instead selected by the network's intermediate representations and are thus well suited for the robot's navigation task. Millán [37] implemented a connectionist RL scheme on a mobile robot learning navigation in office corridors based on dead-reckoning. The approach utilizes several methods for improving the learning rate, including a coarse codification, or generalization, of the sensory inputs, a hard-wired set of basic reflexes in situations of incorrect generalization, a modular network, and constrained search of the action space.

Very few examples of multi-robot learning have been demonstrated so far. Matarić [31] demonstrated learning higher-level group behaviors such as foraging by selecting among basis primitives, the work on which this paper is based. Parker [40] implemented a non-RL memory-based style of parameter-learning for adjusting activation thresholds used to perform task allocation in a multi-robot system. Tan [47] has applied traditional RL to a simulated multi-agent domain. Due to the simplicity of the simulated environment, the work has relied on an MDP model that was not applicable to this domain. Furthermore, Tan [47] and other simulation works that use communication between agents rely on the assumption that agents can correctly exchange learned information. This often does not hold true for physical systems whose noise

and uncertainty properties extend to the communication channels.

In these and most other domains in which RL has been applied, the learning agent attempts to acquire an effective policy for individual (greedy) payoff. In contrast, this paper addresses the problem of learning social rules that allow for optimizing global payoff, but may not "trickle down" to the individuals. This is a particularly challenging form of the credit assignment problem: not only is credit (reward) from the environment delayed, but in many cases of social behavior, it is non-existent. Consequently, other sources of reward, such as social reinforcement, need to be introduced in order to make social rules learnable.

The problem of credit assignment in a group has been addressed in game theory (for example see [2,12,20]) but it has largely been treated under the *rational agent* assumption. According to traditional definitions from game theory, economics, and distributed artificial intelligence (DAI), rational agents are capable of correctly evaluating the utility of their actions and strategies. Much work has been done in opponent modeling and strategy learning for two-agent systems [5,14,38,39,43] and some in the field of DAI on multi-agent Q-learning [16,44].

In situated multi-agent domains, due to incomplete or non-existent world models, inconsistent reinforcement, noise and uncertainty, the agents cannot be assumed to be rational. In general, systems treated by game theory are usually simpler and more cleanly constrained than those found in biology and robotics. This paper focuses on studying what is required for learning social behaviors in domains where the agents cannot be assumed to be rational.

3. Interaction vs. interference

3.1. Resource vs. goal competition

Interference is an unavoidable aspect of multi-agent interaction and is one of the primary motivators for the formation of social rules. We define *interference* as any influence that opposes or blocks an agent's goal-driven behavior. In societies consisting of agents with similar goals, interference largely manifests itself as competition for shared resources. In diverse societies, where agents' goals differ, more complex conflicts can persist between agents, including deadlocks, oscillations, and undoing of one another's work. Social structure serves to minimize interference and maximize the efficiency of the group.

Two functionally distinct types of interference are relevant to this work: interference caused by the multiplicity of agents, which we will call *resource competition*, and interference caused by goal-related conflict, which we will call *goal competition*. Resource competition includes any interference resulting from multiple agents competing for common resources such as space, food, and information. This type of interference causes the decline in performance in multi-agent systems as more agents are added. Goal competition includes any interference resulting from multiple agents having different and potentially conflicting goals.

While resource competition is caused by physical coexistence, and can thus arise in any multi-agent system, goal competition is particularly acute in systems with heterogeneous agents. Functionally different agents can create lasting interference and undo each other's work either out of an immediate need for resources or due to other goals such as, for instance, establishing dominance. For example, a group of agents collecting food from the same source and taking it to the same home experience resource competition. However, two subgroups with different home locations competing over food can experience goal competition as well since the agents of one group could find an incentive for blocking the progress of the members of the other group, as well as for "stealing" the food from their home.

Goal competition is studied primarily by the DAI community (e.g., [13]) whose approaches usually involve predicting other agents' goals and intentions, thus requiring agents to maintain models of each other's internal state (e.g., [18,36], and others mentioned in the related work section). Such prediction abilities require sensory, representational, and communication capabilities and computational resources that typically scale poorly with increased group sizes. In contrast, our work deals with homogeneous societies whose social rules are shared by all individuals. Consequently, only resource competition, and social rules aimed at minimizing it, are relevant and will be addressed.

3.2. Individual vs. group payoff

Social rules that minimize interference among agents attempt to direct behavior away from individual greediness and toward global efficiency. Greedy individualist strategies perform poorly in group situations where inevitable resource competition, which grows with the size of the group, is incorrectly managed. [3] Not all group dynamics fall into this category. For example, some tasks allow for efficient greedy strategies in which agents specialize by task division (for example see demonstrations in [11]). In contrast, this paper focuses on tasks and solutions in which the agents cannot specialize, and instead must find means of optimizing their activity within the same task by developing social rules.

In such situations, agents must give up individual optimality in favor of collective efficiency. At least in theory, it is in the interest of each of the individuals to obey social rules, since on the average their individual efficiency will be improved as well. However, since the connection between individual and collective benefit is not always direct, the problem of learning social rules is a difficult one.

Outside game theory, this problem has been addressed in the field of Artificial Life. Genetic algorithms allow generations of agents with different social rules to be tried out, mutated, and recombined, in order to find the best fit [17,23–25,42]. The work presented in this paper is fundamentally different because it addresses the problem of learning social rules by each of the individuals during their lifetime and within the context of a task, in this case foraging. It is aimed at loosely modeling higher animals rather than insects that are pre-programmed with the appropriate social strategies.

3.3. Prototypical states in social interaction

In theory, the number of possible social states, i.e., states that involve interactions between two or more agents, grows with the size of the group, and the associated rules of social conduct could potentially grow at the same rate. In practice, however, social rules are largely independent of the exact group size. Specifically, only a few group sizes (e.g., two, three, several, many) and prototypical relations among agents (e.g, one-to-one, one-to-few, and one-to-many [4] with the first two being the most prevalent), are relevant for any given type of interaction.

The canonical form of social relation is the *dominance hierarchy* or *pecking order*, ubiquitous in animal societies, from hermit crabs and chickens to primates and people [7,8]. Some biological data support the hypothesis that the number of levels in dominance hierarchies is bounded and relatively stable across a large spectrum of species [8], possibly suggesting that only a small computational/cognitive overhead may be required. Besides mating, the majority of animal/social interaction focuses on establishing and maintaining such pecking orders [9]. A direct evolutionary benefit resulting from them is hard to prove, but hierarchies certainly serve to ritualize and thus simplify social interactions.

While dominance hierarchies are a prevalent social structure and simplify social interaction, in this work we focus on learning social rules that are not directly embedded in a dominance relation. We study rules that are social in that they are derived by the agent based on its interactions with others and their effects on its efficiency over time. This paper uses the group size classification above to prune the state space of social interactions, and focus on learning social rules that are applied in one-to-one and one-to-few interaction classes. In particular, we study learning yielding rules for one-to-one motion conflicts, and communication rules for one-to-few interaction. We postulate that learning social rules for such interactions requires specific types of social reinforcement, as described in the following section.

4. Social reinforcement

Social learning is the process of acquiring new behavior patterns in a social context, by learning from conspecifics. Also called *observational learning*, it is ubiquitous in nature and the propensity for it appears to be innate [34,35]. Social learning includes learning *how* to perform a behavior, through imitation, and

[3] The rate of growth is determined by the properties of the particular system.

[4] Where "many" is bounded by the sensing and communication range.

when to perform it, through social facilitation. *Imitation* is defined as the ability to observe and repeat the behavior of another animal or agent while *social facilitation* refers to the process of selectively expressing a behavior which is already a part of the agent's repertoire. This work is in the latter category, since our agents learn social rules, i.e., when to engage in various built-in social behaviors.

Social settings offer a plethora of information useful for social learning. Observed behavior of conspecifics can serve as negative as well as positive reinforcement. For example, animals quickly learn not to eat food that has had a poisonous effect on others, and to avoid those that have been dangerous to other members of the group [10,15,35]. The so-called vicarious learning, or learning through observation of other agents' experiences, is a means of distributing trials so that one agent need not perform them all. As long as the experience of an agent is "visible", it can serve as a source of vicarious learning trials and social reinforcement signals for others.

Evidence from ethology guides us to propose three forms of reinforcement involved in social learning. The first type is the individual perception of progress relative to the current goal. This type of reinforcement is inherent in most adaptive agent-learning tasks, but its availability varies depending on the specific agent and environment. In most tasks, the agent can maintain some measure of progress that is critical for efficient learning [31].

The second type of reinforcement comes from observing the behavior of conspecifics. Similar behavior in a peer is considered to be a source of positive reinforcement. Although this in itself does not constitute a reliable reinforcement signal, coupled with a direct estimate of progress toward the agent's own goal, it can provide useful feedback.

The third type of reinforcement is that received by conspecifics, also called vicarious reinforcement. Interestingly, using this type of information does not require the agent to model another agent's internal state. If the agents belong to a society with consistent social rules, any reward or punishment received by a conspecific will be received by the agent itself in a similar situation. Consequently, vicarious reinforcement can serve as an effective learning signal.

We postulated that all three of the above described forms of reinforcement are necessary for learning social rules in our domain, and possibly in other, qualitatively similar learning scenarios. Individual reinforcement alone, while effective for learning specific tasks within a group (e.g., foraging [32]), is not sufficient for learning social rules because it, by definition, maximizes individual benefit. In contrast, many social rules do not immediately and directly benefit the individual and, in most cases, have a delaying effect on individual reinforcement. Therefore, since a greedy agent maximizes individual reward and social behaviors may not provide any, learning social rules appears to require a non-greedy approach.

We introduce the impetus for non-greediness through the second and third types of social reinforcement above, namely observing and repeating the actions of others, and receiving vicarious reinforcement. Observing other agents' behavior encourages the agent to explore, i.e., try behaviors that may not benefit it immediately. Repeating the behavior of others enforces multiple trials of what may be an otherwise rare social behavior, so it can receive more stable reinforcement.

Using the three forms of reinforcement allows a fully distributed society to acquire globally optimizing social rules from individual learning, i.e., without a central arbiter. The underlying assumptions are: (1) that the agents are able to estimate other agents' reinforcement, and (2) that their own reinforcement is positively correlated with that of their conspecifics. In short, what is good for one, is good for another, at least indirectly. This simple model allows for learning a variety of powerful social rules that minimize interference and maximize group benefit in a homogeneous society.

To test our ideas, we designed a collection of experiments on situated agents (mobile robots) learning four social rules: *yielding, proceeding, communicating*, and *listening*, in order to become more globally efficient at foraging. The rest of the paper describes the experiments, starting with the experimental environment.

5. Experimental setup

The experimental environment is designed for performing a variety of group behavior experiments. It has been used for verifying work on developing basis behaviors [32] and for demonstrating learning to

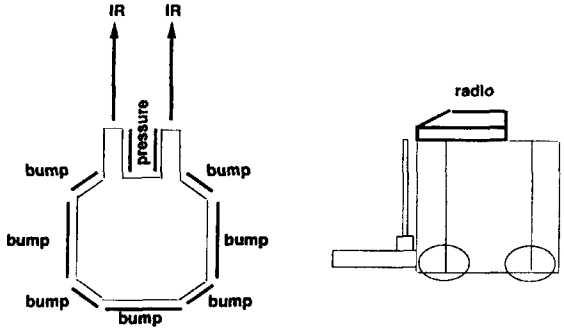


Fig. 1. Each robot consists of a differentially steerable wheeled base and a gripper that can grasp and lift objects. The robots' sensory capabilities include piezoelectric bump and gripper sensors, infrared sensors for collision detection, proprioceptive sensors of drive and gripper-motor current, voltage, and position, and a radio transmitter for communication, absolute positioning, and data collection.

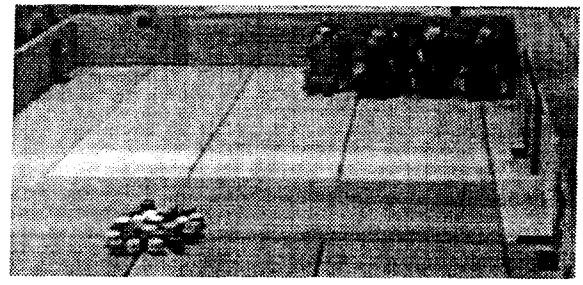


Fig. 2. The testing area for the learning experiments. The figure shows the typical initial condition of a learning trial, with all the robots starting out in the home region, and the food being initially clustered elsewhere in the workspace.

forage [31]. The setup allows for implementing various interactions between robots capable of communicating with each other, and sensing and manipulating objects.

The experiments are conducted on a collection of up to four IS Robotics R2 mobile robots. The robots are fully autonomous with on-board power and sensing. Each robot consists of a differentially-steerable wheeled base and a gripper that can grasp and lift objects. The robot's sensory capabilities include a piezo-electric bump sensor strip around the base for detecting collisions, another strip inside each finger of the gripper for detecting the grasping force, and a set of infra-red (IR) sensors: two on the inside of the fingers for detecting graspable objects, and 10 more for obstacle avoidance: a set of eight sensors around the top of the robot, and one more on each gripper tip (see Fig. 1). In addition to the described external sensors, the robots are also equipped with proprioceptive sensors supplying battery voltage and current information, sensors for drive motor current, and shaft encoders in the gripper motors for maintaining position and height information.

Finally, the robots are equipped with radio transceivers, used for determining absolute position (for the purposes of collecting data as well as for enabling the robots to find and return to the food cluster) and for inter-robot communication. Position information is obtained by triangulating the distance computed from synchronized ultrasound pulses from two fixed base stations. Inter-robot communication consists of locally broadcasting 6-byte messages at the rate of 1 Hz. The radios are particularly useful for transmitting any information that could not be reasonably sensed with the available on-board sensors, such as the external state of other robots (i.e., holding food, finding home, etc.), required for social learning. The robots are programmed in the Behavior Language, a parallel programming language based on the Subsumption Architecture [3,4]. Their control systems are collections of parallel, concurrently active behaviors.

We tested the learning approach on robots situated in a foraging task, having a common higher-level goal of collecting food and taking it home during the "day" and sleeping at home at "night". Thus, the robots' world is a confined area which contains a fixed home region shared by all robots, and scaled so as to accommodate them, i.e., large enough for all of them to "park" for the night (Fig. 2). Thus, the experimental environment is meant to loosely resemble a society that spends its days foraging (hunting and gathering) by making repeated trips to a resource-rich area, getting the food, and taking it home. At fixed periods (meant to resemble night time), the society gathers at home and rests. Foraging activities resume at the beginning of each day.

Food, represented by small metal pucks, is initially clustered in a single location in the workspace (Fig. 2). This initial condition was chosen for two reasons. First, clustering provides an incentive for the agents to cooperate and exchange information in order to locate the single food source. Second, clustering reduces

Condition	Behavior
Finding food	Broadcast
Receiving a message	Store location
Near a stopped agent	Proceed
Near a moving agent	Store behavior
Too near a stopped agent	Proceed
Too near a moving agent	Yield

Fig. 3. The condition-behavior pairings of the desired social policy.

the overall duration of each experimental trial since it significantly diminishes the time spent searching for food. Finite resources such as battery power are conserved by designing the experiment so as to maximize the relevance of the robots' interactions without diminishing the complexity of the learning task. For the same reason, the chosen workspace is small enough to induce frequent interaction and interference between the robots.

6. Learning task and approach

The learning task consisted of the agents acquiring social rules for yielding and proceeding when appropriate, and communicating puck location and listening to received communication when appropriate. These social behaviors result from six social rules, i.e., six condition-behavior pairs (Fig. 3). Yielding consists of learning when to, on one hand, give way and, on the other, keep going, depending on the distance to the other agent and on whether it is stopped or moving. Sharing information consists of learning when to, on one side, broadcast position information and when to, on the other, receive and store it.

Social rules are expressed within the robots' natural habitat and the context of their usual routines, in this case foraging. Foraging was chosen because our previous work [29,30] provided the basis behavior repertoire to which social rules could easily be added. The built-in foraging behavior consists of a finite state controller which, in response to mutually-exclusive conditions, consisting of internal state, activates appropriate basis behaviors from the robots' repertoire. The conditions include externally-triggered events, such as getting close to another robot or finding a puck, and internally-generated events, such as the onset of night time, indicated by the internal clock (since the robots had no external light sensors). Basis behaviors include avoidance, dispersion, searching for pucks, picking up pucks, homing, and sleeping. Our earlier work [31] has shown how foraging itself can be learned, through the use of shaped reinforcement in the form of *multimodal reward functions* that pooled asynchronous reinforcement from all available sources upon termination of a behavior, and *progress estimators* that provided some feedback during the execution of a behavior. The work described here demonstrates how group foraging can be made more efficient with social rules.

Since in our system the agents were learning when to apply the social behaviors, their built-in behavioral repertoire included those social behaviors, along with others giving the robots basic safe navigation and puck manipulation functionalities. Adding social behavior to the system did not require adding new abilities to the robots, since they already contained the necessary actions for each: yielding consists of stopping and waiting, and proceeding consists of going on with the current behavior. The other two behaviors, communicating and listening, were an extension of the existing mechanisms for sending and receiving messages used by the robots to communicate their positions to each other. In the social case, in addition to sending and receiving messages, the robots also stored the received messages and used their contents later, i.e., went to the received location in search of food.

The social learning algorithm is activated whenever an agent finds itself:

(i) near a large amount of food away from home;
(ii) receiving an agent's message;
(iii) within observing range of another stopped agent;
(iv) within observing range of another moving agent;
(v) within interference range of another stopped agent;
(vi) within interference range of another moving agent.

The first condition enables learning to *communicate* about sharable resources such as food. The last two conditions are based on two distance thresholds established a priori: σ_{observe} and $\sigma_{\text{interfere}}$. In the foraging behavior, the presence of another agent within $\sigma_{\text{interfere}}$ triggers avoidance. In the social learning algorithm, a social behavior is attempted as an alternative.

These conditions are specified by the designer in order to speed up the learning. They could be learned using one of the available statistical methods for state generalization (for example, [6,27]) but the process could take almost as long as the social learning and is likely to suffer from sensory errors.

6.1. Learning algorithm

The job of the learning algorithm was to correlate appropriate conditions for each of these behaviors in order to optimize the higher-level behavior, i.e., to maximize received reinforcement. The learning task could be idealized into a simple search in the space of possible strategies, with highest reward for the strategies that result in highest efficiency of foraging. However, in the described domain, agents cannot simply "search" the condition-behavior space because they cannot directly and completely control events in the world. Their world is stochastic, noisy and uncertain, and is made more complex by the existence of other learning agents, whose existence and interactions constitute the relevant conditions for learning social rules.

The system learns the value function $A(c, b)$, expressed as a matrix that associates values for each social behavior b and condition c. Maintaining such a matrix is reasonable, since the number of social conditions (conditions involving the interaction of two or more agents) and social behaviors (in our case, four) is small. The values in the correlation matrix (integers in our implementation), fluctuate over time based on the received reinforcement, but the learning algorithm eventually converges to a stable value function. At each point the value of $A(c, b)$ is the sum of all past reinforcement R:

$$A(c, b) = \sum_{t=1}^{T} R(t).$$

6.2. Implementing social reinforcement

We used the following three types of social reinforcement:

(1) direct reinforcement;
(2) observation of the behavior of other agents;
(3) observation of reinforcement received by other agents.

The implementation of direct reinforcement is straightforward. A progress monitoring behavior constantly compares the agent's current state with its immediate goal. Whenever it detects progress (whether in terms of reaching a subgoal, as in the case of finding food, or in terms of diminishing the distance toward the goal, as in the case of going home), it gives a small reward to the most recently active social behavior.[5] Analogously, whenever a regress from the goal is detected, a small punishment is delivered. Formally

$$D(t) = \begin{cases} m & \text{if progress is made,} \\ n & \text{if regress is made,} \\ 0 & \text{otherwise,} \end{cases} \quad m > 0, \quad n < 0.$$

This algorithm keeps the learning system from settling in local minima, since the system continuously adapts to the condition-action values with each received reinforcement. The algorithm relies on estimating progress at least intermittently. If progress measurements are not available, and the reward function is an impulse at the goal only, then the algorithm reduces to one-step temporal differencing [46], which has been proven to converge, however slowly. In our case intermittent reinforcement can be obtained, so learning is sped up. We used the same method to acquire the basic, greedy strategy for individual foraging [31].

The motivation for using progress estimators (also called internal critics), rather than delayed reinforcement, comes from the non-stationary, uncertainty, and inconsistency properties of the situated multi-robot domain. Progress estimators provide unbiased, principled reinforcement which is not so delayed as to become useless in a dynamic environment (for details, see [31]).

Observational reinforcement, i.e., reinforcement for repeating another agent's behavior, is delivered in a similar form:

$$O(t) = \begin{cases} o & \text{if observed behavior is repeated,} \\ 0 & \text{otherwise,} \end{cases}$$

$o > 0$.

An agent receives positive reinforcement when it repeats a behavior b most recently observed under condition c, next time it finds itself under condition c,

[5] Note that only *social* behaviors are reinforced, since only social rules are being learned.

unless it has already recently performed b. The last part of the rule prevents the agent from repeatedly attempting the same behavior. $O(t)$ has a temporal component. It expires after a fixed time so that the agent, in effect, forgets what it last saw if it has not seen it recently. This feature eliminates some cyclic behavior patterns between multiple learning agents observing each other.

Finally, vicarious reinforcement, i.e., reinforcement received based on that received by other agents, is delivered in the following form:

$$V(t) = \begin{cases} v & \text{if vicarious positive reinforcement,} \\ w & \text{if vicarious negative reinforcement,} \\ 0 & \text{otherwise} \end{cases}$$

for $v > 0, \quad w < 0$.

Vicarious reinforcement delivers a form of "shared" reinforcement to all agents involved in a local social interaction. By spreading individual reward or punishment over multiple agents, it extends individual benefit beyond a single agent. As a consequence, the amount of reward received for social behaviors over time outweighs that received for greedy strategies.

The complete reinforcement function then, is the sum of the subset of social reinforcement being used in the given learning experiment. We tested the following reinforcement functions:

(i) $A_d(c, b) = \sum_{t=1}^{T} D(t)$,

(ii) $A_{do}(c, b) = \sum_{t=1}^{T} (\alpha D(t) + \beta O(t))$, $\alpha + \beta = 1$ and $\alpha > \beta$,

(iii) $A_{dov}(c, b) = \sum_{t=1}^{T} (\alpha D(t) + \beta O(t) + \gamma V(t))$, $\alpha + \beta + \gamma = 1$ and $\alpha > \beta$ and $\beta >= \gamma$.

We used a simple scheme in which direct progress is weighted the highest, while observation-induced experience and vicarious reinforcement are weighted less.

6.3. Experimental design

In a typical learning experiment all the agents are endowed with identical basis behaviors and the social learning reinforcement function. They start out from home at the beginning of the day and go about their foraging task. The social learning algorithm is activated whenever an event occurs that makes one of the six social conditions true. At that point the current behavior the agent is executing is terminated and appropriate social reinforcement is delivered. Then, a new behavior is selected using the following strategy:

(i) choose a recently observed behavior, if available (30%);

(ii) choose an untried social behavior, if available (30%) otherwise;

(iii) choose "the best" behavior (35%);

(iv) choose a random behavior (5%).

Note that after all social behaviors have been explored, the exploration/exploitation ratio in the learner changes so that it performs "the best" behavior 65% of the time, randomly explores 5% of the time, and imitates the rest of the time.

Given the confined physical work area, agent interactions generate a plethora of events enabling social conditions. Consequently, we can observe learning in real time over periods of minutes.

7. Results

As a control experiment, we tested the performance of a pre-programmed foraging behavior that contained the desired social rules, and compared it to base case foraging. Not surprisingly, groups using social rules consistently outperformed groups with only greedy individual strategies, measured in terms of total time required to pick up 80% of the pucks. Thus, we were convinced that the incentive for learning social rules did exist, and it was then a matter of finding out which reinforcement strategy makes it learnable in the given environment.

The relative effectiveness of the three tested reinforcement functions $A_d(c, b)$, $A_{do}(c, b)$, and $A_{dov}(c, b)$ was evaluated based on the portion of the desired social policy the algorithms learned. Fig. 3 shows the condition-behavior pairings of the desired social policy.

The evaluation metrics used to compare the results of the different reinforcement were:

(i) the percentage of the desired social strategy the agents managed to learn;

(ii) the relative amount of time required to learn it.

The duration of any learning run varies depending on the arrangement of the agents and the temporal distribution of their interactions. Consequently, exact time to convergence was not a valid evaluation metric in an event-driven situated environment of the kind

Reinforcement	Performance
$A_d(c, b)$	Does not converge
$A_{do}(c, b)$	Does not converge
$A_{dov}(c, b)$	Converges

Fig. 4. Comparative performance of different reinforcement functions.

Condition	Behavior
Near a stopped agent	Proceed
Too near a stopped agent	Proceed
Receiving a message	Store location
Near a moving agent	Store behavior
Too near a moving agent	Yield
Finding food	Broadcast

Fig. 5. The relative difficulty of learning each condition-action pair, measured in time to convergence, and shown in increasing order from top to bottom.

we are using. Relative average time, i.e., performance relative to the other alternatives, gives us a more useful metric of the comparative effectiveness of different algorithms.

Fig. 4 shows the results in which convergence is defined as learning the complete desired social policy, as shown in Fig. 3. The results shown above are averaged over multiple trials and are qualitative because insufficient trials were run to perform accurate statistical analysis. However, results from the performed trials were consistent in that the first two strategies never converged, and learned only a very small part of the policy. The third strategy converged in over 90% of the trials. It required between 20 and 25 min; in trials that were terminated early (e.g., due to low battery power), the complete desired social policy was not learned. Trial duration was not an issue in case of the other two reinforcement strategies, as they did not improve after learning about 20% of the desired policy (i.e., two rules), which typically took no more than 10 min. Trials were run up to 30 min but the first two reinforcement strategies uniformly failed to converge regardless of experiment duration.

Duration of learning was a direct effect of using physical hardware to run the learning experiments, since the domain was ridden with unavoidable intermittent error and noise. For instance, agents did not always behave consistently due to their inability to accurately sense external state. Such unavoidable (and often externally undetectable) errors generated "unintentional deserters" in that robots experiencing sensor errors (and in some cases radio transmission delays in communication) failed to behave socially due to perceptual errors and noise. While these effects slowed down the learning, they did not disable it because the learning algorithm, based on continuous reinforcement summing, effectively averages out the consequences of intermittent errors.

We considered a rule to have been learned if:

(i) it was correct, i.e., it was the desired mapping between the given condition and the associated behavior;

(ii) it was stable, i.e., it did not oscillate or switch to another (incorrect) condition-behavior association.

Fig. 5 shows the ordering of the social rules in terms of the learning difficulty, based on the average time required to learn a given rule per trial. This ordering reflects the immediacy of the reward associated with each social behavior. The condition-behavior pairs that produce the most immediate reward, i.e., those that involved proceeding around a stopped agent, were learned the fastest. In the case of the first two reinforcement schemes, they were also the only ones to be learned.

The social rules involving proceeding when near and too near a stopped agent are relatively easy to learn since the reinforcement for making progress toward the individual goal location (a pile of pucks or home) is immediate. The underlying obstacle avoidance behavior keeps the robots from colliding with the stopped robots, and the social rule simply reinforces what may be considered to be a natural, basis, behavior.

In contrast, learning to yield when near a moving agent is more difficult to learn since the payoff is much more delayed, and arrives vicariously, through another agent. The yielding agent must learn to perform a behavior that involves a suboptimal use of its individual resources, but benefits the group as a whole by minimizing collective interference. However, since the agent has no direct measure of group interference,

it must wait for the indirect reward. Consequently, this social rule is the second hardest to learn.

The social rule of sharing information about food was the hardest to learn, since the agent sending the information had the least direct payoff. Consequently, multiple instances of observational reinforcement were needed to instigate the exploration of the behavior. After broadcasting food location, the agent often receives no immediate vicarious reinforcement because the other agent, who received the message, has not yet learned to store and use the information. Once the agent tries to store the information and use it (i.e., go to the transmitted location and find food), it receives positive reinforcement, and passes it on to the agent who originally broadcast the information. This already indirect process is frequently further delayed since using the transmitted information can only be useful to a robot that is not already carrying a puck and is already going toward a known puck location. To conserve the transmitted information, the robots remember the latest received message; when a robot finishes dropping off a puck, it looks up the stored communication message (if it has any) and proceeds to the specified location. The logic behind keeping just the latest message is grounded in the intuition that it was the most likely to be accurate. Although the pucks are initially clustered, they quickly become displaced through manipulation, so the communicated puck locations can become outdated.

Social rules dealing with communication, which requires the agent to store either the location or the observed behavior, are relatively easy to learn. They took slightly longer than the rules for proceeding because the feedback was somewhat delayed. Specifically, a proceeding agent receives positive feedback immediately, since it continues to get closer to its destination. An agent storing information will receive feedback when it successfully uses that information, which could happen almost immediately, or with some delay, but almost always faster than the indirect process involved in rewarding broadcasting. The difference lies in the fact that the reward for broadcasting must come from other agents, while the reward for storing/remembering is received directly when the information is used, i.e., when the agent finds the food or tries a successful behavior.

8. Discussion

This qualitative analysis of the results provides an intuitive explanation of the relative difficulty of the attempted social rules in terms of immediacy and directness of reinforcement. However, more experiments with different tasks and domains are required to test the validity of this explanation. We expect that in more complex learning systems other factors, such as the relative importance and other semantic values of the reinforcement, may have an equally strong or stronger effect.

The intermediate process required for learning to communicate could have been simplified if agents were learning from a teacher. However, we were interested in having social rules emerge in a homogeneous group, so we only used imitation to the extent of mimicking observed behavior by a peer, not a selected expert or teacher. Our continuing work explores learning by imitation in this and other domains.

The difficulty of learning yielding and broadcasting gives us a hint of the challenges involved in acquiring non-greedy social rules in situated domains. It is likely that learning particularly challenging altruistic rules whose payoff is only at the genetic level (e.g., predator signaling), indicates that those may themselves be best acquired genetically. Data from biology seem to support this intuition, since animals do not appear to learn altruism toward their kin but do learn social dominance relations [34].

One frequently asked question is whether this work could not just as easily (or much more easily) have been done in simulation. We believe that simple simulations can serve as useful tools for preliminary testing of control and learning algorithms. However, for predicting the effectiveness of such algorithms on physical systems, it is necessary to model all the features of the physical systems that would impact the resulting behavior. Unfortunately, we do not know a priori what those features are, and we do know that the always relevant features that involve sensor and effector characteristics are very challenging to simulate accurately. Thus, we believe that it is important to validate the proposed learning approaches on systems that most closely model the target systems for which they are being postulated and developed. For our work, mobile robots are the best such model.

9. Summary and continuing work

This work has focused on learning social rules in situated multi-agent domains. We have studied the difficulty of learning behaviors which do not offer direct benefit to the agent and are in contradiction with its basic, greedy, survival instincts. We postulated three types of reinforcement that are useful and possibly necessary for learning such social rules. We then tested three reinforcement combinations by applying an already effective situated learning algorithm to the social learning problem and adding the proposed types of reinforcement to it. We demonstrated the algorithms by implementing them on a group of four autonomous mobile robots capable of communication and cooperation and given the task of learning yielding and sharing information.

The results presented are only a glimpse at the wide variety of social rules that can be learned, and forms of social reinforcement that are worth exploring. In order to properly evaluate our theories, we continue to implement more in-depth experiments. We are interested in expanding this work in a number of directions. In particular, it would be interesting to consider variations on the learning experiments, such as a gathering task with multiple food and home regions, in order to study what kinds of specializations emerge between agents and how these affect the resulting social rules. We would also like to test the given social reinforcement strategies on quite different types of tasks in order to see how general they are. Another area we are interested in exploring is learning to distinguish what aspects of the situation (state) are relevant, in the context of learning relevant group sizes and dominance hierarchies. If this turns out to be difficult to learn it will give us an idea of what types of biases may be genetically programmed.

In addition to the foraging-based experiment, we have also worked with learning in the box-pushing domain, where two or more agents must learn to coordinate their actions in order to successfully achieve the task. Unlike foraging, which can be achieved with a single agent, the pushing experiments require tight cooperation and sharing of the agents' limited local sensory and effector resources [33]. At least one social rule or convention, turn-taking, was involved in our experiments in learning cooperative box-pushing [45]. We are interested in exploring such tightly coupled cooperative tasks as a domain for studying social rules.

The goal of the work presented has been to provide insights into how difficult certain types of social learning may be and how we may go about successfully synthesizing them on situated embodied agents.

Acknowledgements

The research reported here was done partly at the MIT Artificial Intelligence Laboratory, and subsequently continued at the author's Interaction Laboratory at Brandeis University, in the Volen Center for Complex Systems and the Computer Science Department. Work done at MIT was supported in part by the Jet Propulsion Laboratory and in part by the Advanced Research Projects Agency under the Office of Naval Research. Work done at Brandeis is supported by the Office of Naval Research Grant N00014-95-1-0759 and the National Science Foundation Infrastructure Grant CDA-9512448.

The author thanks the three anonymous reviewers and in particular Rolf Pfeifer for detailed, insightful, and constructive comments.

References

[1] M. Asada, E. Uchibe, S. Noda, S. Tawaratsumida and K. Hosoda, Coordination of multiple behaviors acquired by a avision-based reinforcement learning, *Proc. IEEE/RSJ/GI Int. Conf. on Intelligent Robots and Systems*, Munich, Germany (1994).

[2] R. Axelrod, *The Evolution of Cooperation* (Basic Books, NY, 1984).

[3] R.A. Brooks, A hardware retargetable distributed layered architecture for mobile robot control, *IEEE Int. Conf. on Robotics and Automation*, Raleigh, NC (1987) 106–110.

[4] R.A. Brooks, The behavior language; User's guide, Technical Report, AIM-1227, MIT Artificial Intelligence Lab, 1990.

[5] D. Carmel and S. Markovitch, Opponent modeling in multi-agent systems, in: G. Weiss and S. Sen, eds., *Adaptation and Learning in Multi-Agent Systems*, Lecture Notes in Artificial Intelligence, Vol. 1042 (Springer, Berlin, 1996) 40–52.

[6] D. Chapman and L.P. Kaelbling, Input generalization in delayed reinforcement learning: An algorithm and performance comparisons, *Proc. IJCAI-91*, Sydney, Australia (1991).

[7] I.D. Chase, Dynamics of hierarchy formation: The sequential development of dominance relationships, *Behaviour* 80 (1982) 218–240.

[8] I.D. Chase and S. Rohwer, Two methods for quantifying the development of dominance hierarchies in large groups with application to Harris' sparrows, *Animal Behavior* 35 (1987) 1113–1128.

[9] D.L. Cheney and R.M. Seyfarth, *How Monkeys See the World* (The University of Chicago Press, Chicago, 1990).

[10] J.M. Davis, Imitation: A review and critique, in: Bateson and Klopfer, eds., *Perspectives in Ethology*, Vol. 1 (Plenum Press, New York, 1973).

[11] J.L. Deneubourg, S. Goss, J.M. Pasteels, D. Fresneau and J.P. Lachaud, Self-organization mechanisms in ant societies, II: Learning in foraging and division of labor, *From Individual to Collective Behavior in Social Insects* 54 (1987) 177–196.

[12] E.H. Durfee, J. Lee and P.J. Gmytrasiewicz, Overeager reciprocal rationality and mixed strategy equilibria, in: *Proc. AAAI-93*, Washington, DC (1993) 225–230.

[13] L. Gasser and M.N. Huhns, *Distributed Artificial Intelligence* (Pitman, London, 1989).

[14] C.V. Goldman and J.S. Rosenschein, Mutually supervised learning in multiagent systems, in: G. Weiss and S. Sen, eds., *Adaptation and Learning in Multi-Agent Systems*, Lecture Notes in Artificial Intelligence, Vol. 1042 (Springer, Berlin, 1996) 85–96.

[15] J.L. Gould, *Ethology; The Mechanisms and Evolution of Behavior* (Norton, NY, 1982).

[16] P. Gu and A.B. Maddox, A framework for distributed reinforcement learning, in: G. Weiss and S. Sen, eds., *Adaptation and Learning in Multi-Agent Systems*, Lecture Notes in Artificial Intelligence, Vol. 1042 (Springer, Berlin, 1996) 97–112.

[17] T. Haynes and S. Sen, Evolving behavioral strategies in predators and prey, in: G. Weiss and S. Sen, eds., *Adaptation and Learning in Multi-Agent Systems*, Lecture Notes in Artificial Intelligence, Vol. 1042 (Springer, Berlin, 1996) 113–126.

[18] M.J. Huber and E.H. Durfee, Observational uncertainty in plan recognition among interacting robots, in: *Proc. IJCAI-93 Workshop on Dynamically Interacting Robots*, Chambery, France (1993) 68–75.

[19] L.P. Kaelbling, Learning in embedded systems, Ph.D. thesis, Stanford University, 1990.

[20] S. Kraus, Agents contracting tasks in non-collaborative environments, in: *Proc. AAAI-93*, Washington, DC (1993) 243–248.

[21] L.-J. Lin, Programming robots using reinforcement learning and teaching, in: *Proc. AAAI-91*, Pittsburgh, PA (1991) 781–786.

[22] L.-J. Lin, Self-improving reactive agents: Case studies of reinforcement learning frameworks, in: *From Animals to Animats: Int. Conf. on Simulation of Adaptive Behavior* (MIT Press, Cambridge, MA, 1991).

[23] H.H. Lund, Specialization under social conditions in shared environments, in: *Proc. Advances in Artificial Life, 3rd European Conf. on Artificial Life (ECAL)* (1995) 477–489.

[24] B.J. MacLennan, Evolution of communication in a population of simple machines, Technical Report, Computer Science Department, CS–90–104, University of Tennessee, 1990.

[25] B.J. MacLennan and G.M. Burghardt, Synthetic ecology and the evolution of cooperative communication, *Adaptive Behavior* 2 (2) (1994) 161–188.

[26] P. Maes and R.A. Brooks, Learning to coordinate behaviors, in: *Proc. AAAI-91*, Boston, MA (1990) 796–802.

[27] S. Mahadevan and J. Connell, Automatic programming of behavior-based robots using reinforcement learning, *Proc. AAAI-91*, Pittsburgh, PA (1991) 8–14.

[28] S. Mahadevan and J. Connell, Scaling reinforcement learning to robotics by exploiting the subsumption architecture, *Proc. 8th Int. Workshop on Machine Learning* (Morgan Kaufmann, Los Altos, CA, 1991) 328–337.

[29] M.J. Matarić, Designing emergent behaviors: From local interactions to collective intelligence, in: J.-A. Meyer, H. Roitblat and S. Wilson, eds., *From Animals to Animats: Int. Conf. on Simulation of Adaptive Behavior.*

[30] M.J. Matarić, Kin recognition, similarity, and group behavior, *Proc. 15th Annual Conf. on Cognitive Science Society*, Boulder, Colorado (1993) 705–710.

[31] M.J. Matarić, Reward functions for accelerated learning, in: W.W. Cohen and H. Hirsh, eds., *Proc. 11th Int. Conf. on Machine Learning (ML-94)*, (Morgan Kauffman, New Brunswick, NJ, 1994) 181–189.

[32] M.J. Matarić, Designing and understanding adaptive group behavior, *Adaptive Behavior* 4 (1) (1995) 50–81.

[33] M.J. Matarić, M. Nilsson and K.T. Simsarian, Cooperative multi-robot box-pushing, *Proc. IROS-95* (IEEE Computer Society Press, Los Alamitos, CA, 1995).

[34] D. McFarland, *Animal Behavior* (Benjamin Cummings, Menlo Park, CA, 1985).

[35] D. McFarland, *The Oxford Companion to Animal Behavior* (Oxford University Press, Oxford, 1981).

[36] M. Miceli and A. Cesta, Strategic social planning: Looking for willingness in multi-agent domains, *Proc. 15th Annual Conf. on Cognitive Science Society*, Boulder, Colorado (1993) 741–746.

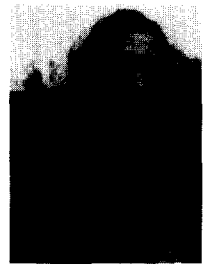
[37] J.D.R. Millán, Learning reactive sequences from basic reflexes, *Proc. Simulation of Adaptive Behavior, SAB-94* (MIT Press, Brighton, 1994) 266–274.

[38] Y. Mor, C.V. Goldman and J.S. Rosenschein, Learn your opponent's strategy (in polynomial time)!, in: G. Weiss and S. Sen, eds., *Adaptation and Learning in Multi-Agent Systems*, Lecture Notes in Artificial Intelligence, Vol. 1042 (Springer, Berlin, 1996) 164–176.

[39] T. Ohko, K. Hiraki and Y. Anzai, Learning to reduce communication cost on task negotiation among multiple autonomous mobile robots, in: G. Weiss and S. Sen, eds., *Adaptation and Learning in Multi-Agent Systems*, Lecture Notes in Artificial Intelligence, Vol. 1042 (Springer, Berlin, 1996) 177–190.

[40] L.E. Parker, Heterogeneous multi-robot cooperation, Ph.D. thesis, MIT, 1994.

[41] D.A. Pomerleau, Neural network perception for mobile robotic guidance, Ph.D. thesis, Carnegie Mellon University, School of Computer Science, 1992.
[42] *Proceedings, Artificial Life II* (Addison-Wesley, Reading, MA, 1989).
[43] T.W. Sandholm and R.H. Crites, On multiagent Q-learning in a semi-competitive domain, in: G. Weiss and S. Sen, eds., *Adaptation and Learning in Multi-Agent Systems*, Lecture Notes in Artificial Intelligence, Vol. 1042 (Springer, Berlin, 1996) 191–217.
[44] S. Sen and M. Sekaran, Multiagent coordination with learning classifier systems, in: G. Weiss and S. Sen, eds., *Adaptation and Learning in Multi-Agent Systems*, Lecture Notes in Artificial Intelligence, Vol. 1042 (Springer, Berlin, 1996) 218–233.
[45] K.T. Simsarian and M.J. Matarić, Learning to cooperate using two six-legged mobile robots, *Proc. 3rd European Workshop of Learning Robots*, Heraklion, Crete, Greece (1995).
[46] R. Sutton, Learning to predict by method of temporal differences, *Machine Learning* 3 (1) (1988) 9–44.
[47] M. Tan, Multi-agent reinforcement learning: Independent vs. cooperative agents, *Proc. 10th Int. Conf. on Machine Learning*, Amherst, MA (1993) 330–337.
[48] S.B. Thrun and T.M. Mitchell, Integrating inductive neural network learning and explanation-based learning, *Proc. IJCAI-93*, Chambery, France (1993).



Maja J. Mataric is an assistant professor in the Computer Science Department and the Volen Center for Complex Systems at Brandeis University near Boston. She received a Ph.D. in computer science and artificial intelligence from MIT in 1994. She has worked at NASA's Jet Propulsion Lab, the Free University of Brussels AI Lab, LEGO Cambridge Research Labs, GTE Research Labs, the Swedish Institute of Computer Science, and ATR. Her Interaction Laboratory at Brandeis conducts research on the dynamics of interaction in complex adaptive systems including multi-agent systems ranging from a group of up to 26 mobile robots to economies and ecologies. Her current work covers the areas of control and learning in intelligent situated agents, and cognitive neuroscience modeling of visuo-motor skill learning through imitation.