Multi-Agent Reinforcement Learning: Independent vs. Cooperative Agents

Ming Tan

GTE Laboratories Incorporated
40 Sylvan Road
Waltham, MA 02254
tan@gte.com

Abstract

Intelligent human agents exist in a cooperative social environment that facilitates learning. They learn not only by trial-and-error, but also through *cooperation* by sharing instantaneous information, episodic experience, and learned knowledge. The key investigations of this paper are, "Given the same number of reinforcement learning agents, will cooperative agents outperform independent agents who do not communicate during learning?" and "What is the price for such cooperation?" Using independent agents as a benchmark, cooperative agents are studied in following ways: (1) sharing sensation, (2) sharing episodes, and (3) sharing learned policies. This paper shows that (a) additional sensation from another agent is beneficial if it can be used efficiently, (b) sharing learned policies or episodes among agents speeds up learning at the cost of communication, and (c) for joint tasks, agents engaging in partnership can significantly outperform independent agents although they may learn slowly in the beginning. These tradeoffs are not just limited to multi-agent reinforcement learning.

1 INTRODUCTION

In human society, learning is an essential component of intelligent behavior. However, each individual agent need not learn everything from scratch by its own discovery. Instead, they exchange information and knowledge with each other and learn from their peers or teachers. When a task is too big for a single agent to handle, they may cooperate in order to accomplish the task. Examples are common in non-human societies as well. For example, ants are known to communicate about the locations of food, and to move objects collectively.

In this paper, I use reinforcement learning to study intelligent agents (Mahadevan & Connel 1991, Lin 1991, Tan 1991). Each reinforcement-learning agent can incrementally learn an efficient decision policy over a state space by trial-and-error, where the only input from an environment is a delayed scalar reward. The task of each agent is to maximize the long-term discounted reward per action.

Although most work on reinforcement learning has focused exclusively on single agents, we can extend reinforcement learning straightforwardly to multiple agents if they are all independent. They together will outperform any single agent due to the fact that they have more resources and a better chance of receiving rewards. Recently, Whitehead (1991) has also demonstrated the potential benefit of multiple "complete-observing" cooperative agents over a single agent. However, the more practical study is to compare the performance of n independent agents with the one of n cooperative agents and to identify their tradeoffs. Yet, no such study has been done previously. It is the subject of this paper.

How can reinforcement-learning agents be cooperative? I identify three ways of cooperation. First, agents can communicate instantaneous information such as sensation, actions, or rewards. Second, agents can communicate episodes that are sequences of (sensation, action, reward) triples experienced by agents. Third, agents can communicate learned decision policies. This paper presents three case studies of multi-agent reinforcement learning involving such cooperation and draws some related conclusions that are not limited to multi-agent reinforcement learning. The main thesis of this paper is that *if cooperation is done intelligently, each agent can benefit from other agents' instantaneous information, episodic experience, and learned knowledge.*

Specifically, in case study 1, I investigate the ability of an agent to utilize sensation input provided by another agent. I demonstrate that sensory information from another agent is beneficial only if it is relevant

and sufficient for learning. I show one instance where cooperative agents were not able to efficiently learn decision policies (compared with independent agents) due to insufficient sensation from other agents.

Case study 2 focuses on sharing learned policies and episodes. I show that in these cases cooperation speeds up learning, but does not affect asymptotic performance. I also provide upper bounds on their communication costs incurred during cooperation. While sharing policies is limited to homogeneous agents, sharing episodes can be used by heterogeneous agents as long as they can interpret episodes.

Case study 3 concerns joint tasks which require more than one agent in order to be accomplished. I demonstrate that cooperative agents who sense their partners or communicate their sensations with each other can learn to perform the tasks at a level that independent agents cannot reach even though they start out slowly. If a cooperative agent must sense other agents, the size of its state space can increase exponentially in terms of the number of involved agents.

Ideally, intelligent agents would learn when to cooperate and which cooperative method to use to achieve maximum gain. This paper is a starting point for the examination of these fundamental open questions.

2 RELATED WORK

Several multi-agent learning systems have been developed for speed and/or accuracy. GTE's ILS system (Silver et. al 1990) integrates heterogeneous (inductive, search-based, and knowledge-based) learning agents by a central controller through which the agents critique each other's proposals. The MALE system (Sian 1991) uses an interaction board (similar to a blackboard) to coordinate different learning agents. DLS (Shaw & Sikora 1990) adopts a distributed problem-solving approach to rule induction by dividing data among inductive learning agents. Recently, Chan and Stolfo (1993) advocate meta-learning for distributed learning. Most of these systems deal with inductive learning from examples, rather than autonomous learning agents that involve perception and action. One exception to this is the complexity analysis of cooperative mechanisms in reinforcement learning by Whitehead (1991). His main theorem is that n reinforcement-learning agents who can observe everything about each other can decrease the required learning time at a rate that is $\Omega(1/n)$.

Recent work in the field of *Distributed Artificial Intelligence* (DAI) (Gasser & Huhns 1989) has addressed the issues of organization, coordination, and cooperation among agents, but not for multi-agent learning. In the terms of DAI, my case studies 1 and 2 explore reinforcement learning in *collaborative reasoning systems* (Pope et. al 1992) which are concerned with coordinating intelligent behavior across multiple self-sufficient agents, and my case study 3 studies reinforcement learning in *distributed problem-solving systems* (Durfee 1988, Tan & Weihmayer 1992) in which a particular problem is divided among agents that cooperate and interact to develop a solution. Unlike DAI, this work does not deal with issues such as communication language, agent beliefs, resource constraint, and negotiation. It also mainly focus on homogeneous agents.

3 REINFORCEMENT LEARNING

Reinforcement learning is an on-line technique that approximates the conventional optimal control technique known as *dynamic programming* (Bellman 1957). The external world is modeled as a discrete-time, finite state, Markov decision process. Each action is associated with a reward. The task of reinforcement learning is to maximize the long-term discounted reward per action.

In this study, each reinforcement-learning agent uses the one-step *Q-learning* algorithm (Watkins 1989). Its learned decision policy is determined by the state/action value function, Q, which estimates long-term discounted rewards for each state/action pair. Given a current state x and available actions a_i, a Q-learning agent selects each action a with a probability given by the Boltzmann distribution:

$$p(a_i|x) = \frac{e^{Q(x,a_i)/T}}{\sum_{k \in actions} e^{Q(x,a_k)/T}} \qquad (1)$$

where T is the temperature parameter that adjusts the randomness of decisions. The agent then executes the action, receives an immediate reward r, moves to the next state y.

In each time step, the agent updates $Q(x,a)$ by recursively discounting future utilities and weighting them by a positive learning rate β:

$$Q(x,a) \leftarrow Q(x,a) + \beta(r + \gamma V(y) - Q(x,a)) \qquad (2)$$

Here γ $(0 \leq \gamma < 1)$ is a discount parameter, and $V(x)$ is given by:

$$V(x) = \max_{b \in actions} Q(x,b) \qquad (3)$$

Note that $Q(x,a)$ is updated only when taking action a from state x. Selecting actions stochastically by (1) ensures that each action will be evaluated repeatedly.

As the agent explores the state space, its estimate Q improves gradually, and, eventually, each $V(x)$ approaches: $E\{\sum_{n=1}^{\infty} \gamma^{n-1} r_{t+n}\}$. Here r_t is the reward received at time t due to the action chosen at time $t-1$. Watkins and Dayan (1992) have shown that this Q-learning algorithm converges to an optimal decision policy for a finite Markov decision process.

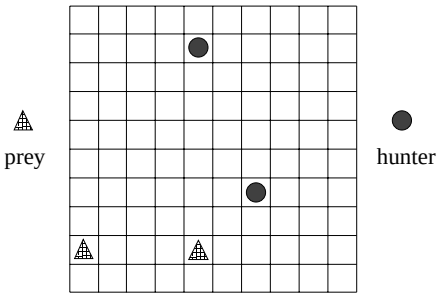
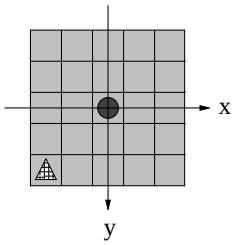


Figure 1: A 10 by 10 grid world.



A perceptual state represented by (-2, 2)

Figure 2: A visual field of depth 2.

4 TASK DESCRIPTION

All the tasks considered in this study involve hunter agents seeking to capture randomly-moving prey agents in a 10 by 10 grid world, as shown by Figure 1. On each time step, each agent (hunter or prey) has four possible actions to choose from: moving up, down, left, or right within the boundary. Initially, hunters also make random moves as they have equal Q values. More than one agent can occupy the same cell. A prey is captured when it occupies the same cell as a hunter (in case study 1 and 2) or when two hunters either occupy the same cell as the prey or are next to the prey (in case study 3). Upon capturing a prey, the hunter or hunters involved receive $+1$ reward. Hunters receive -0.1 reward for each move when they do not capture a prey. Each hunter has a limited visual field inside which it can locate prey accurately. Figure 2 shows a visual field of depth 2. Each hunter's sensation is represented by (x, y) where x (y) is the relative distance of the closest prey to the hunter according to its x (y) axis. For example, (-2, 2) is a perceptual state when the closest prey is in the lower left corner of the hunter's visual field (see Figure 2). If two prey are equally close to a hunter, only one of them (chosen randomly) will be sensed. If there is no prey in sight, a unique default sensation is used.

Each run of each experiment consisted of a sequence of trials. In the first trial of each run, all agents were given a random location. Afterwards, each trial began with only rewarded hunters in random locations. Each trial ended when the first prey was captured. Each run was given a sufficient number of trials until the decision policies of hunters converged (i.e., the performance of hunters stabilized). I measured the average number of time steps per trial in *training* where actions were selected by the Boltzmann distribution, at intervals of every 50 trials. After convergence, I also measured the average number of time steps per trial in *test* where actions were selected by the highest Q value, over at least 1000 trials. Results were averaged over at least 5 runs.

The Q-learning parameters were set at $\beta = 0.8$, $\gamma = 0.9$, and $T = 0.4$. These values are reasonable for these tasks. Task parameters include the number of prey, the number of hunters, and the hunters' visual-field depth.

Without learning, hunters move randomly with baseline performances for four different prey/hunter tasks given in Table 1. The table shows the average number of steps for random hunters to capture a prey over 200 trials. I also tested the performances of independently learning hunters for the corresponding tasks. Table 1 gives their average number of steps to capture a prey in training calculated after a sufficient number of trials, where the hunters' visual-field depth was 4. Clearly, learning hunters significantly outperform random hunters. The real question is whether or not cooperation among learning hunters can further improve their performance.

5 CASE 1: SHARING SENSATION

First, I study the effect of sensation from another agent. To isolate sensing from learning, I choose the one-prey/one-hunter task and add a scouting agent that cannot capture prey. Later I extend this concept to hunters that perform both scouting and hunting. I demonstrate that sensory information from another (scouting) agent is beneficial if the information is relevant and sufficient for learning.

The scout makes random moves. At each step, the scout send its action and sensation back to the hunter. Assume that the initial relative location between the scout and the hunter is known. Therefore, the hunter can incrementally update the scout's relative location and also compute the location of the prey sensed by the scout. For example, if the relative locations of a prey to the scout (known) and the scout to the hunter (sensed) are (-2, 2) and (2, 5) respectively, then the relative location of the prey to the hunter is (0, 7). To keep the same dimension of a state representation (i.e., still use (x, y)), I combine sensation inputs from the hunter and the scout as follows: use the hunter's sensation first, if the hunter cannot sense any prey, then use the scout's sensation.

Table 2 shows the average numbers of steps to capture

Table 1: Average Number of Steps to Capture a Prey: Random vs. Independently Learning Hunters.

N-of-prey/N-of-hunters	1/1	1/2	1/2 (joint task)	2/2 (joint task)
Random hunters	123.08	56.47	354.45	224.92
Learning hunters	25.32	12.21	119.17	100.61

Table 2: Scouting vs. No Scouting.

Hunter Visual Depth	Scout Visual Depth	Average Steps to Capture a Prey	
		Training	Test
2	no scouting	47.14 ($\pm$1.28)	49.49 ($\pm$1.60)
2	2	46.33 ($\pm$1.39)	42.91 ($\pm$1.48)
2	3	39.78 ($\pm$1.06)	32.08 ($\pm$1.22)
2	4	32.67 ($\pm$1.03)	25.07 ($\pm$0.89)

a prey in training after 2000 trials and the ones in test after convergence with or without a scout. Their 90% confidence intervals calculated by a *t-test* are listed in the parentheses. The hunter with a scout took fewer steps in both training and test to capture a prey than the one without.[1] As the scout's visual-field depth increases, the difference in their performances becomes larger. This observation held when the hunter's visual-field depth was given other values (other than 2). Based on this state representation, the maximum number of perceptual states in the 10 by 10 grid world is 442 ($= (2\times10+1)^2+1$). After introducing a scout, the size of the state space for the hunter was effectively increased from 26 ($= 5^2 + 1$) to 442. This increase was traded for extra sensory information and paid off in the end. In fact, when the scout's visual-field depth was 4, no obvious slowdown was observed after only 50 trials.

Once establishing the benefit of additional sensory information from a scout, I then extended this concept to the one-prey/two-hunter task with each hunter acting as a scout for the other hunter. Table 3 gives the similar measures for both independent and mutual-scouting agents. Their 90% confidence intervals calculated by a *t-test* and the resulting *t-test* comparisons within each pair are given in the parentheses. As their visual-field depth increases, (a) both independent and mutual-scouting agents take fewer and fewer steps to capture a prey; (b) mutual-scouting agents gradually outperform independent agents; and (c) the advantage of mutual-scouting agents over independent agents shows up sooner in test than in training. As an example, when the visual-field depth was 4, mutual-scouting hunters took, on the average, 8.83 steps in test to capture a prey comparing with 11.53 steps for independent hunters. However, when the visual-field depth was limited to 2, sharing sensory information hindered *training*, because a short-sighted scouting hunter could not stay with a prey long enough for the other hunter to learn to catch up with the prey. This suggests that sensory information from another agent should be used prudently, and extra, insufficient information can interfere with learning. Scouting also incurs communication cost. The information communicated from a mutual-scouting agent to another agent per step is bounded by the size (in bits) of its sensation and action representation. In this experiment, it is $2\log_2(2V_{depth}+1)+2$ where V_{depth} is the visual-field depth.

[1] Although the average steps of the hunter in training with a scout whose visual-field depth was 2 ($= 46.33$) is less than the one of the hunter without a scout ($= 47.14$), the difference is not significant according to the *t-test*.

6 CASE 2: SHARING POLICIES OR EPISODES

Assume that agents do not share sensation. If each agent is adequate to accomplish a task (e.g., each hunter can capture a prey by itself), is cooperation among agents still useful? I studied several ways of sharing learned policies and episodes in the one-prey/two-hunter task. Hunters can either (1) use the same decision policy or (2) exchange their individual policies at various frequencies. Episodes can be exchanged (a) among peer hunters or (b) between peer and expert hunters. I will show that such cooperative agents can speed up learning, measured by the average number of steps in training, even though they will eventually reach the same asymptotic performance as independent agents. This study presents the experimental results when the hunters' visual-field depth is

Table 3: Two Independent Agents vs. Two Mutual-Scouting Agents.

	Visual Depth	Average Steps to Capture a Prey	
		Training	Test
Independent agents	2	20.38 (±0.57)	24.04 (±1.00)
Mutual-scouting agents	2	25.20 (±0.79) (worse)	24.52 (±1.24) (same)
Independent agents	3	14.65 (±0.53)	16.04 (±0.56)
Mutual-scouting agents	3	14.02 (±0.75) (same)	12.98 (±0.65) (better)
Independent agents	4	12.21 (±0.65)	11.53 (±0.61)
Mutual-scouting agents	4	11.05 (±0.56) (better)	8.83 (±0.78) (better)

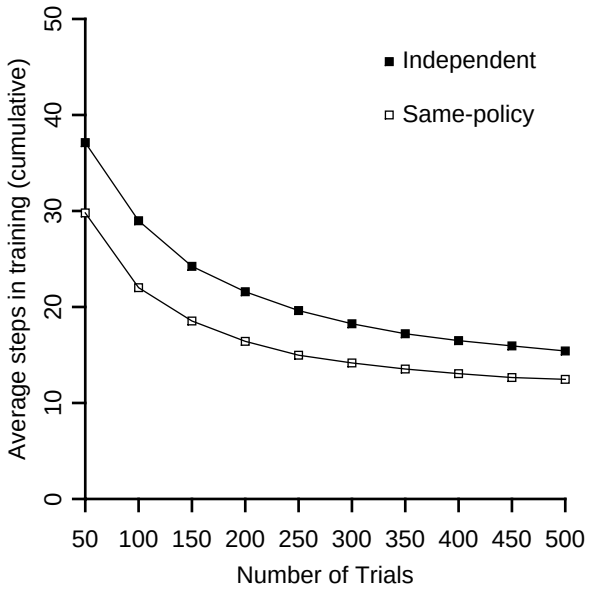


Figure 3: Independent agents vs. same-policy agents.

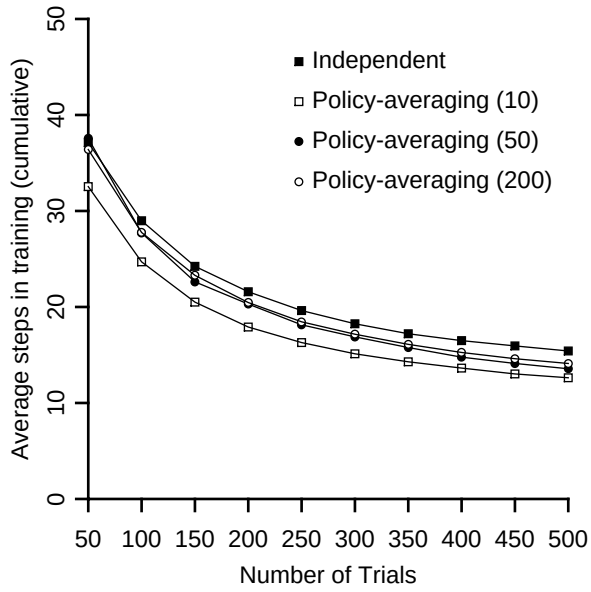


Figure 4: Independent agents vs. policy-averaging agents.

4. The conclusions when the visual-field depth is 2 or 3 are similar to 4.

One simple way of cooperating is that hunters use the same decision policy. Although each hunter updates the same policy independently, the rate of updating the policy is multiplied by the number of hunters per step. Figure 3 shows that when two hunters used the same policy, they converged much quicker than two independent hunters did. The average information communicated by each same-policy hunter per step is bounded by the number of the bits needed to describe a sensation, an action and a reward.[2] In this experiment, it is $2\log_2(2V_{depth}+1)+3$.

[2]I assume that only one agent keeps a decision policy. At each step, the rest of the involved agents send their current sensation to the policy-keeping agent, receive corresponding actions in return, and then send the rewards of their actions back to the policy-keeping agent.

If agents perform the same task, their decision policies during learning can differ because they may have explored the different parts of a state space. Two hunters can complement each other by exchanging their policies and use what the other agent had already learned for its own benefit. Assume that each agent can simultaneously send its current policy to other agents, I adopted the following policy assimilation: agents average their policies at certain frequency. Figure 4 shows the performance results when two hunters averaged their policies at every 10 steps, 50 steps, or 200 steps. All of them converged quicker than two independent hunters. One interesting observation is that when the visual-field depth was 4, the best frequency was every 10 steps (see Figure 4) while when the visual-field depth was 2, the best frequency was every 50 steps (not shown here). In general, the information communicated by each policy-exchanging hunter per step is bounded by $(N-1)\cdot P\cdot F$ where N is the number

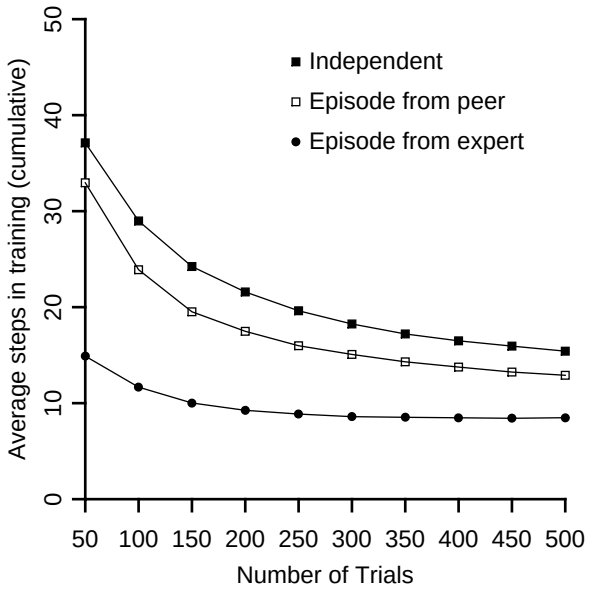


Figure 5: Independent agents vs. episode-exchanging agents.

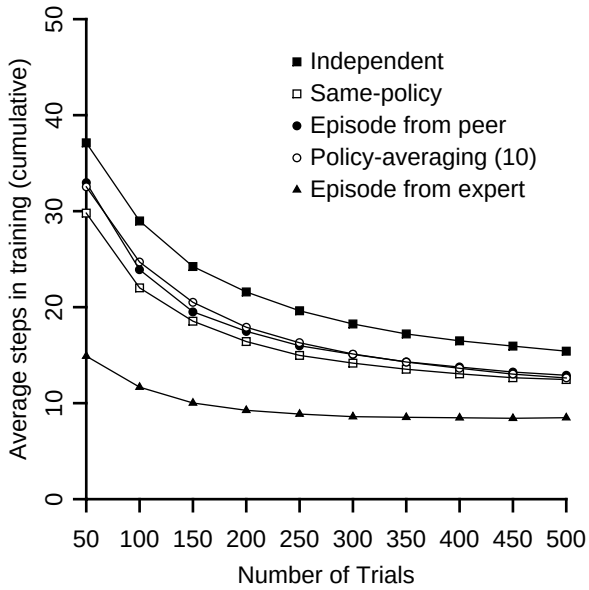


Figure 6: Summary.

of participating hunters, P is the size of a policy (i.e., number of perceptual states $\times$ number of actions $\times$ number of bits needed to represent a sensation, an action and a Q value), and F is the frequency of policy exchanging. When P or F is large, communication can be costly. On the other hand, unlike same-policy agents, a policy-exchanging agent can be selective in assimilating another agent's policy. For example, an agent could adopt another agent's decision only when it did not have confidence in certain actions.

Instead of sharing learned knowledge such as a policy, agents can share their episodes. An episode is a sequence of (sensation, action, reward) triples experienced by an agent. I used the following episode exchanging: when a hunter captured a prey, the hunter transferred its entire solution episode to the other hunter. The other hunter then "mentally replayed" the episode forward to update its own policy. As a result, two hunters doubled their learning experience. The middle curve in Figure 5 shows the speedup in training of two hunters after exchanging their episodes. The average information communicated by each episode-exchanging hunter per step is bounded by $(N-1) \cdot E$ where E is the number of bits needed to represent a sensation, an action, and a reward ($E = 2\log_2(2V_{depth}+1)+3$ in this experiment). In addition to the flexibility of assimilating episodes, exchanging episodes can be used by heterogeneous reinforcement-learning agents as long as they can interpret episodes (e.g., hunters can have different visual-field depths). To demonstrate this point, I let two hunters learn from an expert hunter that always moves towards the prey using the shortest path. Figure 5 shows significant improvement for the two novice hunters when the episodes they received were from an expert hunter (see the bottom curve). Note that an expert hunter could be just another hunter who has already learned hunting skills. This result demonstrates another benefit of learning in a cooperative society where novices can learn quickly from experts by examples (Lin 1991, Whitehead 1991).

Figure 6 summarizes the experimental results of this case study. Generally speaking, during the early phase of training, cooperative learning outperforms independent learning, and learning from an expert outperforms both. Their differences in performance are statistically significant according to *t-tests*. However, among different ways of cooperation (excluding learning from an expert), there is no conclusive evidence that one performs better than the others. In terms of the average information communicated, if the number of participating agents is limited to 2, exchanging episodes is comparable to using the same policy. Exchanging policy is plausible if the size of a policy is small and the proper frequency of policy exchanging can be determined.

7 CASE 3: ON JOINT TASKS

In the previous two case studies, each hunter can capture prey by itself. Here, I study joint tasks where a prey can only be captured by two hunters who either occupy the same cell as the prey as or are next to the prey. Hunters cooperate by either passively observing each other or actively sharing their sensations and locations. I demonstrate that cooperative agents can learn to perform the joint task significantly better than independent agents although they start slowly.

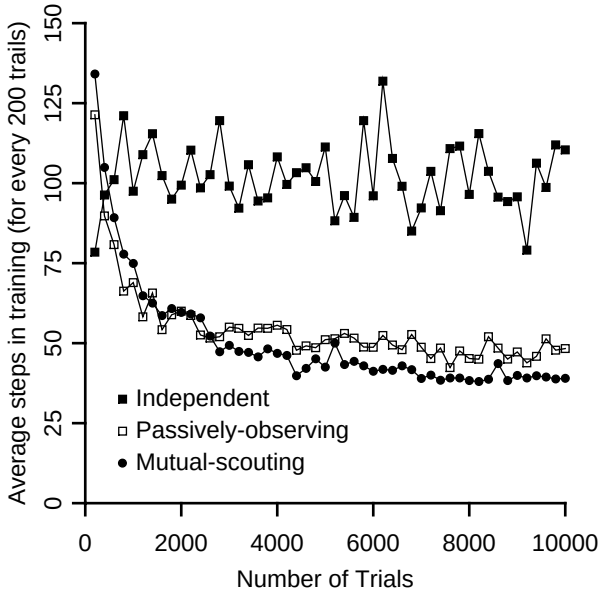


Figure 7: Typical runs for the 2-prey/2-hunter joint task.

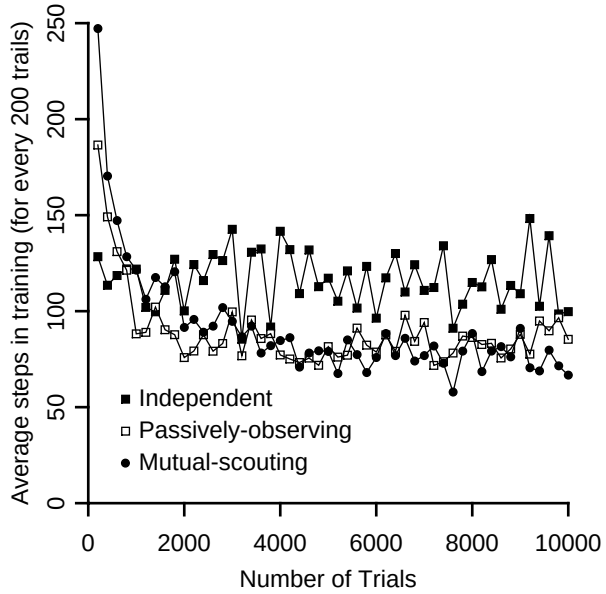


Figure 8: Typical runs for the 1-prey/2-hunter joint task.

Assume that the hunters' visual-field depth is 4 (again, the conclusions are similar when the visual-field depth is 2 or 3). Let us first consider the two-prey/two-hunter joint task. When two independent hunters were given this task, each hunter tended to learn to approach a prey directly. When both hunters approached the same prey, they succeeded and received rewards. When they chased two different prey, they failed and were penalized. As training continued, their performance fluctuated noticeably around the level of taking, on the average, 101 steps to capture a prey (see the top curve in Figure 7).

The problem with independent hunters is that they ignore each other. They cannot distinguish the situation where another hunter is nearby from the one far away. If each hunter can also sense the other hunter, cooperative behavior can emerge from greedy learning hunters. To address this problem, I extended the sensation of a hunter to two pairs $\{(x_{prey}, y_{prey})(x_{ptn}, y_{ptn})\}$ where (x_{prey}, y_{prey}) is the relative location ($\leq$ visual-field depth) between a prey and the hunter, and (x_{ptn}, y_{ptn}) between a partner and the hunter. Note that the state space is increased exponentially in terms of the number of agents. A large state space means more state exploration for a hunter, and slower learning. Nevertheless, although starting slowly, such passively-observing hunters began to overtake independent hunters soon after 400 trials, and eventually reduced the average number of steps to only 49 (see the middle curve in Figure 7).

Two hunters can cooperate passively by observing each other in addition to prey. Given the encouraging results from case study 1, I proceeded to let hunters also actively share their sensory information. This means that the state space is further enlarged although there is no increase in the dimension of a state representation. This enlargement made initial learning even slower than passively-observing hunters. Yet, mutual-scouting hunters soon outperformed passively-observing agents after about 1400 trials, and settled down at average 39 steps in training (see the bottom curve in Figure 7). The average number of steps per trial in test for independent, passively-observing and mutual-scouting hunters are 49, 42 and 34, respectively.

People may wonder what would happen if there was only one prey in the joint task. Independent hunters might do well because both hunters can just learn to approach the prey directly. This, however, is not the case. By knowing where its partner is, a hunter can learn better approach (herding) patterns. Figure 8 shows the typical runs of the three types of hunters when there was only one prey. As you can see, independent agents, passively-observing agents, and mutual-scouting agents settled down at average 116, 84, and 76 steps in training, respectively. Although it is difficult to analyze the hunters' specific approach patterns, the fact that cooperative hunters outperformed independent hunters by at least 32 steps per trial suggests the existence of such patterns.

8 CONCLUSIONS AND FUTURE WORK

This paper demonstrates that reinforcement-learning agents can learn cooperative behavior in a simulated social environment. Although this paper's results are

based on simulated prey/hunter tasks, I believe the conclusions can be applied to cooperation among autonomous learning agents in general. This paper identifies three ways of agent cooperation, i.e., by communicating instantaneous information, episodic experience, and learned knowledge. Specifically, cooperative reinforcement-learning agents can learn faster and converge sooner than independent agents via sharing learned policies or solution episodes. Cooperative agents can also broaden their sensation via mutual scouting, and can handle joint tasks via sensing other partners. On the other hand, this paper also shows that extra sensory information can interfere with learning, sharing knowledge or episodes comes with a communication cost, and it takes a larger state space to learn cooperative behavior for joint tasks. These tradeoffs must be taken into consideration for autonomous and cooperative learning agents.

This research raises several important issues of multi-agent reinforcement learning. First, sensation must be selective because the size of a state space can increase exponentially in terms of the number of involved agents. One heuristic used here is that each hunter only pays attention to the nearest prey (or hunter). Can such selective sensation strategies be learned? Second, on a related issue, one needs to use generalization techniques to reduce a state space and improve performance for complex, noisy tasks. Third, learning opportunities are hard to come by for nontrivial cooperative behavior. If a prey were smart enough to know how to escape, it could take a long time for hunters to get enough learning experience. How can learning be more focused (e.g., by learning from a teacher)? Fourth, information exchanging among agents incurs communication costs. Can agents learn to communicate? This learning task gets complicated when the content of communication can be instantaneous information, episodic experience, and learned knowledge. Fifth, other cooperative methods need to be explored. For example, what if hunters share their action intentions to avoid collision, or share their rewards to sustain hunger? Finally, can homogeneous agents learn to have job division and to specialize differently? Can heterogeneous agents (such as scouting agents vs. blind hunting agents) learn to cooperate? These are directions for future work.

Acknowledgments

I am grateful to Rich Sutton, Steve Whitehead, and Chris Matheus for useful discussions and careful comments. I would like to thank Shri Goyal for his support of this research.

References

Bellman, R. E. (1957). *Dynamic Programming.* Princeton University Press, Princeton, NJ.

Chan, P. K. & Stolfo, J. S. (1993). Toward parallel and distributed learning by meta-learning, Proceedings of AAAI Workshop on Knowledge Discovery in Databases, To appear.

Durfee, E. H. (1988). *Coordination of Distributed Problem Solvers*, Kluwer Academic Publishers, Boston.

Gasser, L. & Huhns, M. (1989). *Distributed Artificial Intelligence, 2*, (eds.) Pitman, London.

Lin, L. J. (1991). Programming robots using reinforcement learning and teaching. In Proceedings of AAAI-91. (pp. 781-786).

Mahadevan, S. & Connel, J. (1991). Automatic programming of behavior-based robots using reinforcement learning. In Proceedings of AAAI-91. (pp. 768-773).

Pope, R., Conry, S., & Meyer, R. (1992). Distributing the planning process in a dynamic environment. Proceedings of the 11th International Workshop on Distributed AI, Glen Arbor, MI.

Shaw, M. J. & Sikora, R. (1990). A distributed problem-solving approach to rule induction: learning in distributed artificial intelligence systems. Technical Report, CMU-RI-TR-90-28, The Robotics Institute, Carnegie Mellon University.

Sian, S. S. (1991). Extending learning to multiple agents: issues and a model for multi-agent machine learning. In Y. Kodratoff (Ed.), *Machine Learning – EWSL 91.* Springer-Verlag, pp. 440-456.

Silver, B., Frawely, W., Iba, G., Vittal, J., & Bradford, K. (1990). A framework for multi-paradigmatic learning. In Proceedings of the Seventh International Conference on Machine Learning, 348-358. Austin, Texas.

Tan, M. (1991). Cost-sensitive reinforcement learning for adaptive classification and control. In Proceedings of AAAI-91. (pp. 774-780).

Tan, M. & Weihmayer, R. (1992). Integrating agent-oriented programming and planning for cooperative problem solving. Proceedings of the AAAI-92's Workshop on Cooperation among Heterogeneous Intelligent Agents, San Jose, CA,

Watkins, C. J. C. H. (1989). Learning With Delayed Rewards. Ph.D. thesis, Cambridge University Psychology Department.

Watkins, C. J. C. H. & Dayan, P. (1992) Technical Note: Q-Learning. *Machine Learning*, 8(3/4), Kluwer Academic Publishers.

Whitehead, S. D. (1991). A complexity analysis of cooperative mechanisms in reinforcement learning. In Proceedings of AAAI-91. (pp. 607-613)