Learning Complementary Multiagent Behaviors: a Case Study

(Extended Abstract)

Shivaram Kalyanakrishnan
Department of Computer Sciences
The University of Texas at Austin
shivaram@cs.utexas.edu

Peter Stone
Department of Computer Sciences
The University of Texas at Austin
pstone@cs.utexas.edu

Categories and Subject Descriptors

I.2.6 [**Computing Methodologies**]: Artificial Intelligence—*Learning*

General Terms

Algorithms, Experimentation.

Keywords

Multiagent Reinforcement learning, Policy search, Robot soccer.

1. INTRODUCTION

As the reach of multiagent reinforcement learning extends to increasingly complex tasks, it is likely that the diverse challenges encountered can only be surmounted by combining the strengths of different learning methods. We consider this aspect of learning through the case study of Keepaway, a popular benchmark for multiagent reinforcement learning from the robot soccer domain. Whereas previous successful results in this domain have limited learning to an isolated, infrequent decision that amounts to a turn-taking behavior (Pass), we expand the agents' learning capability to include the more ubiquitous action of moving without the ball (GetOpen), such that at any given time, multiple agents are executing learned behaviors simultaneously. We introduce a policy search method for learning GetOpen to complement the temporal difference learning approach employed for learning Pass [4]. The learned GetOpen policy matches the best hand-coded policy for this task, and outperforms the best policy found when Pass is learned. We demonstrate that Pass and GetOpen can be learned simultaneously, and indeed that these learned behaviors specialize towards the counterpart behaviors with which they are trained.

2. KEEPAWAY Pass AND GetOpen

Keepaway [4] is a subtask in simulated RoboCup soccer [2], in which a team of 3 *keepers* aims to keep possession of the ball away from the opposing team of 2 *takers*. The game is played within a square region of side 20m. Each episode begins with some keeper having the ball, and ends when some taker claims possession or the ball overshoots the region of play. The objective of the keepers is to maximize the expected duration of the episode, called the episodic "hold time". The opposing team of takers seeks to minimize the hold time. Figure 1(a) shows a snapshot of a Keepaway episode in progress.

Figure 1(b) outlines the policy followed by *each* keeper in the scheme employed earlier by Stone *et al.* [4]. The keeper closest to the ball intercepts the ball until it has possession. Once it has possession, it must execute the Pass behavior, by way of which it may retain ball possession or pass to a teammate. Keepers other than the one closest to the ball move to a target position conducive for receiving a pass by executing GetOpen behavior. Most previous work on Keepaway [4] has focused on learning Pass, taking GetOpen to follow a fixed hand-coded strategy. We extend learning in Keepaway to include GetOpen, treating it as a composite of two distinct behaviors to be learned: Pass and GetOpen. This makes Keepaway very challenging compared to previous multiagent learning tasks such as predator-prey [1].

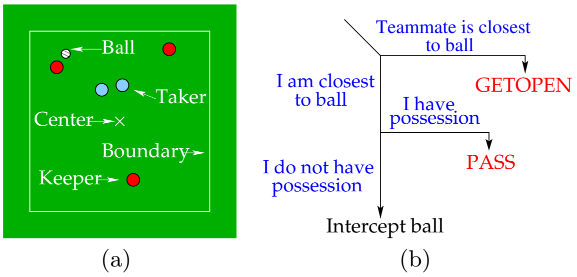

Figure 1: (a) Snapshot from a Keepaway episode. (b) Policy followed by each keeper.

3. LEARNING FRAMEWORK

We apply the algorithm and parameter values employed by Stone *et al.* for learning Pass [4], under which each keeper learns *autonomously* using Sarsa. The reward for a keeper's action is the time elapsed until the next action is taken from the next state by that keeper (or the episode ends). Assuming that the keepers follow stationary GetOpen policies, this scheme directly maximizes the hold time by separately improving each keeper's Pass policy.

In our formulation of GetOpen, keepers have to choose a target point from among 25 points that are uniformly spaced across the playing region. For every target point, we define 10 state variables, including distances and angles involving it and the players. Given an evaluation function over these 10 state variables to assign values to points, and assuming that each keeper will head to the target point yielding the highest

Cite as: Learning Complementary Multiagent Behaviors: a Case Study, (Extended Abstract), Shivaram Kalyanakrishnan, Peter Stone, *Proc. of 8th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2009),* Decker, Sichman, Sierra and Castelfranchi (eds.), May, 10–15, 2009, Budapest, Hungary, pp. 1359–1360

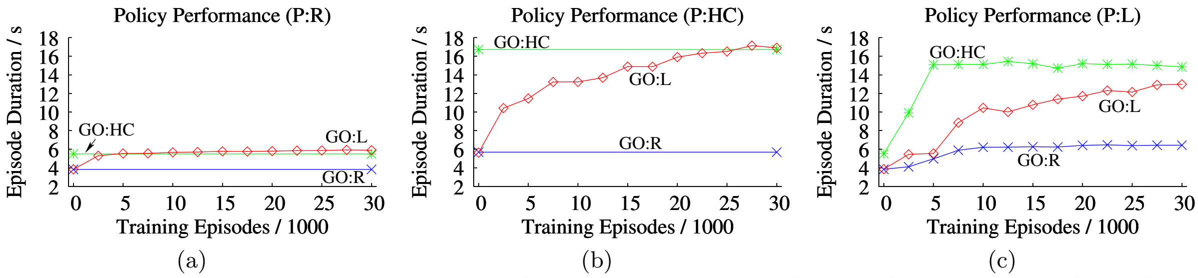

Figure 2: Three GetOpen policies – Random (GO:R), Hand-coded (GO:HC) and learned (GO:L) – when paired with a Pass policy that is (a) Random (P:R), (b) Hand-coded (P:HC), or (c) Learned (P:L).

value, the GetOpen learning problem is to find an evaluation function that maximizes hold time. At any instant of time, two keepers execute GetOpen, having to pick from 25 target points (Pass has three actions). Thus GetOpen induces a more complex MDP than Pass.

We apply a policy search method to optimize the GetOpen policy, which is *shared* by the three keepers. Our learned GetOpen policy is implicitly represented through a neural network that computes a value for a target location given the 10-dimensional input state. We achieve the best results using a 10-5-5-1 network with sigmoid units, with a total of 91 parameters. The policy search method we employ is the cross-entropy method [3]. For learning both Pass and GetOpen, we alternate between learning Pass using Sarsa for a certain number of episodes, during which the GetOpen policy is fixed, after which Pass is fixed and GetOpen is improved using policy search.

4. RESULTS

In our experiments, we consider three variants each of Pass (P)and GetOpen (GO): a random policy (R), a well-tuned hand-coded policy (HC) [4], and the learned policy (L). Figure 2 shows the performance of the nine combinations that arise in total. P:L-GO:HC corresponds to the experiment conducted by Stone *et al.* [4], from which we obtain similar results. After 30,000 episodes of training, the hold time achieved is about 14.9 seconds, which falls short of the 16.7 seconds registered by the static P:HC-GO:HC policy. Interestingly, when paired with the random GetOpen policy GO:R, P:HC is overtaken by P:L at 30,000 episodes ($p < 0.0001$). This result highlights the ability of learning methods to adapt to different settings, for which hand-coded approaches may demand tedious manual attention.

Figure 2 confirms the viability of our policy search method for learning GetOpen and its robustness in adapting to different Pass policies. After 30,000 episodes, P:HC-GO:L achieves a hold time of 16.9 seconds, which indeed exceeds the hold time of P:HC-GO:HC; yet, despite running 20 independent trials of each, this result is not statistically significant. Thus, we only conclude that when coupled with P:HC, learning GetOpen, a novel contribution of this work, matches the hand-coded GetOpen policy that has been used in all previous studies on the Keepaway task. The hold time of P:HC-GO:L is significantly higher than that of P:L-GO:HC ($p < 0.001$). In other words, our GetOpen learning approach outperforms the previously studied Pass learning when each is paired with a hand-coded counterpart, establishing the relevance of learning GetOpen.

In Figure 2(c), we observe that Pass and GetOpen can indeed be learned in tandem. P-L:GO-L achieves a hold time of 13.0 seconds after 30,000 episodes, which is evidence of successful learning, although it falls short of P:L-GO:HC, P:HC-GO:L, and P:HC-GO:HC ($p < 0.001$). Thus, there exists significant scope for improving this result. We conduct a further experiment to ascertain the degree of specialization achieved by learned Pass and GetOpen policies, i.e., whether it is beneficial to learn Pass specifically for a given GetOpen policy (and vice versa). Indeed, we notice that predominantly, the best *learned* Pass policy for a given GetOpen policy is one that was trained with the same GetOpen policy (and vice versa). Many practical problems demand specialized solutions for specific situations; by automatically gravitating towards tightly-coupled behaviors that maximize performance, learning can offer significant gains.

5. CONCLUSION

Through a concrete case study, we advance the case for applying different learning algorithms to qualitatively distinct behaviors present in a complex multiagent system. In particular, we introduce Keepaway GetOpen as a multiagent learning problem that complements Keepaway Pass, the well-studied reinforcement learning test-bed problem from the robot soccer domain. We provide a policy search method for learning GetOpen, which compares on par with a well-tuned hand-coded GetOpen policy, and which can also be learned simultaneously with Pass to realize tightly-coupled behaviors. Showcasing the richness of the Pass+GetOpen learning problem, this work opens numerous avenues for future research.

6. ACKNOWLEDGMENTS

The authors thank Ian Fasel and anonymous reviewers of earlier versions of this paper for useful comments. This work has taken place in the Learning Agents Research Group (LARG) at the Artificial Intelligence Laboratory, The University of Texas at Austin. LARG research is supported in part by grants from the National Science Foundation (CNS-0615104, EIA-0303609 and IIS-0237699), DARPA (FA8750-05-2-0283, FA8650-08-C-7812 and HR0011-04-1-0035), General Motors, and the Federal Highway Administration (DTFH61-07-H-00030).

7. REFERENCES

[1] M. Benda, V. Jagannathan, and R. Dodhiawala. On optimal cooperation of knowledge sources - an empirical investigation. Technical Report BCS–G2010–28, Boeing Advanced Technology Center, Boeing Computing Services, Seattle, WA, July 1986.

[2] M. Chen, E. Foroughi, F. Heintz, Z. Huang, S. Kapetanakis, K. Kostiadis, J. Kummeneje, I. Noda, O. Obst, P. Riley, T. Steffens, Y. Wang, and X. Yin. Users manual: RoboCup soccer server — for soccer server version 7.07 and later. *The RoboCup Federation*, August 2002.

[3] P. T. De Boer, D. P. Kroese, S. Mannor, and R. Rubinstein. A tutorial on the cross-entropy method. *Annals of Operations Research*, 134(1):19–67, 2005.

[4] P. Stone, R. S. Sutton, and G. Kuhlmann. Reinforcement learning for RoboCup-soccer keepaway. *Adaptive Behavior*, 13(3):165–188, 2005.