

Research Goals

Scott Proper

June 5, 2009

1. Introduction

My research goal is to continue to advance the state of the art in multi-agent reinforcement learning. There exist many real world examples of multi-agent domains that I plan to work with in the future, such as fire and emergency response in a wide scale emergency situation (such as an earthquake), vehicle routing and product delivery, passenger pickup, dropoff, and scheduling for taxis, and games such as real-time strategy games with many units on each side of a conflict. All of these domains have something in common: all require multiple agents to coordinate in order to complete multiple tasks.

To solve these multi-agent problems, I plan to apply reinforcement learning. Reinforcement learning is a method of teaching a software agent to learn how to take correct actions in the environment via a process of trial and error. The environment is often described by vector of features which make up the *state* of the environment. One or more agents may take actions in the environment, resulting in a change of state. Agents also receive a *reward* or *cost* signal to indicate when their action was appropriate in that state. Via a *value function*, agents may associate rewards with the particular actions and states that obtained them, and can use dynamic programming to propagate that information. This information may also be used to create a *policy* - or mapping from states to actions - for solving the problem.

As one might expect, this process of reinforcement learning becomes quite difficult as the number of states, agents, and available actions in the environment (or *domain*) increase. This difficulty is sometimes referred to as *the three curses of dimensionality*. First, the state space (and time required for convergence) grows exponentially in the number of state features in the domain. Second, the space of possible actions is exponential in the number of agents, so performing a search over all actions to determine which might be the best is computationally expensive. Lastly, computing the expected value of the next state is costly, as the number of possible future states is exponential in the number of state features.

A variety of techniques are required to overcome the three curses of dimensionality. Most notably, in my prior research I have developed a method called *assignment-based decomposition*. This reinforcement learning technique is designed especially for domains with multiple agents and multiple tasks in a cooperative setting. This technique works by decomposing the action-selection step of any reinforcement learning algorithm into an assignment level that assigns agents to partic-

ular tasks, and a task execution level that chooses actions for agents given their assigned task. This has the consequence of mitigating all three curses of dimensionality. However, assignment-based decomposition is still a very new technique and much research is required to fully develop it so that it might be useful to the broadest range of researchers and research problems, such as those given above. Thus, I anticipate that my research efforts will focus on further improvements to this existing technique and other related methods for scaling reinforcement learning in such a way as to overcome and mitigate the three curses of dimensionality. I expect this work to include research in several broad areas:

2. Decentralized Optimization

It can be argued that most real-world multiagent problems are decentralized: that is, agents have no centralized controller and must communicate via limited channels (such as radio, for emergency or police vehicles). Unfortunately assignment-based decomposition is not currently well-suited to decentralized domains: this technique assumes there is a centralized controller to make assignment decisions.

Some progress has already been made towards a solution. In particular, coordination graphs are a popular technique for resolving conflicts and coordinating between multiple agents. Using techniques such as the decentralized max-plus algorithm, an approximate solution to a coordination graph may be found quickly via message passing. Coordination graphs have so far only been successfully applied to model-free reinforcement learning methods (which do not store an explicit model of the state transition probabilities and rewards). It is not yet clear how coordination graphs may be applied to model-based methods (which store an explicit model of state transitions and rewards), particularly in the context of assignment-based decomposition. I intend to research this.

Coordination graphs are only useful for action selection in assignment-based decomposition when an assignment of agents to tasks has already been found. The problem of decentralized assignment as yet has no solution. However, there does exist much prior work in auction-based methods for solving the assignment problem in competitive domains. Unfortunately very little prior work exists for solving “collaborative auctions”, i.e. auctions in which agents are not actually competing, but are trying to assign agents to tasks in some globally optimal fashion, as in assignment-based decompo-

sition. This literature may offer improved assignment algorithms and should certainly be reviewed and tested against existing assignment algorithms.

3. Improved Assignment Search Techniques

Previous research in fast search techniques has provided several useful methods for quickly assigning agents to tasks, including hill climbing and bipartite search using the Hungarian method. There is a great deal of room for even better search techniques, including:

Mixed Integer Linear Programming (MILP) is a method for optimizing an objective function given linear inequality constraints. Certain variables may be constrained to be integer only. In the case of assignment-based decomposition, we wish to find the maximum value of an assignment, subject to certain constraints. Fast MILP solvers could allow an assignment solution to be found very quickly.

Iterated Bipartite Search: Bipartite search is a method for specifying the relationships of assignments and tasks as a bipartite graph, and using the Hungarian method to find a solution to the graph in polynomial time. Unfortunately this method suffers from the fact that relationships between more than one agent and one task can not be expressed on any particular edge of the graph. One possibility for mitigating this problem is to adjust the values of each edge of the graph after each assignment has been found according to the new information, and using the Hungarian method again. This can be repeatedly tried until convergence (if convergence is possible). This approach needs to be tested.

Learning at the Assignment Level: Assignment-based decomposition is a powerful technique for solving problems involving multiple agents and multiple tasks, but it has its flaws: so far, it has not been shown to solve domains with very different kinds of agents and/or tasks very well. For example, it may not be able to handle cases in which certain tasks should be completed before other tasks for optimal results. To deal with this problem, it will likely be necessary to borrow techniques from hierarchical reinforcement learning by introducing learning at the assignment level. The lower-level value function may be adequate to determine how well a particular set of agents may be suited to complete a particular task, but it will take learning and practice to determine which tasks should be completed first.

4. Automatic Template Learning

Function approximation is a method for mitigating the first curse of dimensionality, or explosion in the number of states, by storing a more compact representation of the value function. Relational Templates are a powerful and flexible function approximation technique that generalize tables, tile coding, and linear functions for relational domains. Relational

templates have a great deal of potential to allow domains using assignment-based decomposition to scale to large sizes, particularly via transfer learning.

There are two broad questions that need to be answered about relation templates: can these templates be learned automatically, and can the process of generalizing between related domains be automated via a similar process? Currently relational templates must be hand-created, and the process of using these templates to generalize requires hand-created “hierarchies” of specializations of templates. If either or both of these processes could be automated using one of a number of machine learning techniques, using relational templates could be much easier, especially in domains where expert knowledge is lacking.

5. Transfer Learning

Transfer learning and generalization are related techniques for transferring knowledge from one domain to another (or alternately, generalizing knowledge from one domain to another). Due to the decomposed nature of assignment-based decomposition, using transfer learning to scale a domain from a small number of agents or tasks to a large number is a natural fit. This has the effect of allowing a policy to be learned in a much smaller setting, and having the possibility of transferring that policy to a larger domain. This mitigates the three curses of dimensionality by bypassing them entirely by simply learning on an easier domain. I intend to research ways in which this may be done most effectively.

A further, more exciting possibility exists for transfer learning and assignment-based decomposition. This is the possibility of learning on one or more sets of different but related domains (for example, Starcraft and/or Warcraft, both real-time strategy games) and transferring that knowledge to another related domain (such as Civilization or Age of Empires). Assignment-based decomposition could assist in this task by observing relationships between similar kinds of agents: for example, archers in Warcraft and marines in Starcraft, and noticing that these relationships are similar to those found in the “target” domain, such as musketeers in Age of Empires. If these relationships can be drawn across many units, it is possible that learning on the target domain could be greatly assisted. I intend to investigate this.