Analyzing Agents that Learn about Agents

José M. Vidal and Edmund H. Durfee

Artificial Intelligence Laboratory, University of Michigan
1101 Beal Avenue, Ann Arbor, MI 48109-2110
jmvidal@umich.edu

A designer of an agent in a multi-agent system (MAS) must decide how much his agent will know about other agents. He can choose to either implement this knowledge directly into the agent, or let the agent learn. For example, he might decide to start the agent with no knowledge and let it learn which actions to take based on its experience, or he might give it knowledge about what to do given the actions of other agents and then have it learn which actions the others' take, or he might give it deeper knowledge about the others (if available), etc. It is not clear which one of the many options is better, especially if the other agents are also learning.

Our research provides a framework for describing the MAS and the different types of knowledge in an agent. We assume that the MAS is discrete, such that agents take an action at each time step and these are all effectively concurrent. We also assume that all agents can see the same world $w \in W$, the actions $a \in A$ taken by all, and have common knowledge of this fact. The agents' knowledge can then be characterized as *nested agent models*. We define a **0-level** agent i as only having knowledge that maps from world states to actions $f_i(w) : a_i$, i.e. $K_i(f_i(w))$. A **1-level** agent is aware of the other agents in the world and views them as 0-level agents, so i knows that j knows a world to action mapping $f_{ij}(w) : a_j$, i.e. $K_iK_j(f_{ij}(w))$. i also knows what action to take given the actions $\vec{a}_{-i}$ taken by others, i.e. $K_i(f_i(w, \vec{a}_{-i}))$. We can keep defining **n-level** agents in a similar way. We find that, if all agents' actions are not impacted by the actions of others, the system will eventually **converge**, i.e. all agents stop changing their behaviors. If a system does converge then the agents can collapse their deeper models into a 0-level model without losing information.

Using our framework, we were also able to calculate the sizes of the hypothesis spaces $|H|$ for the different types of knowledge. With these we can calculate the sample learning complexity, using the standard equation $m \geq \frac{1}{\epsilon}(\ln \frac{|H|}{\gamma})$. This gives us an idea of the number of examples a PAC-learning algorithm would have to see before reaching an acceptable model. We calculated these $|H|$'s for both reinforcement and supervised learning algorithms, as seen in the Table below. Using these results, a designer can determine the expected accuracy of his proposed agent's models. This information is very useful in determining which knowledge should be incorporated and which should be learned by the agent. One interesting result we proved is: In a MAS where $|A|$ is the same for all agents and agents know what to do if they are given the others' action, then a 1-level agent will generally fare better than a 0-level reinforcement learning agent since he will have better models.

	Knowledge	Superv. L.	Reinf. L.
0	$K_i(f_i(w))$	$\lvert A\rvert^{\lvert W\rvert}$	$\lvert R_i\rvert^{\lvert A\rvert\cdot\lvert W\rvert}$
1	$K_i(f_i(w, \vec{a}_{-i}))$	$\lvert A\rvert^{\lvert A\rvert^n}\lvert W\rvert$	$\lvert R_i\rvert^{\lvert A\rvert^n}\lvert W\rvert$
	$K_iK_j(f_{ij}(w))$	$\lvert A\rvert^{\lvert W\rvert}$	$\lvert R_j\rvert^{\lvert A\rvert\lvert W\rvert}$
2	$K_i(f_i(w, \vec{a}_{-i}))$	$\lvert A\rvert^{\lvert A\rvert^n}\lvert W\rvert$	$\lvert R_i\rvert^{\lvert A\rvert^n}\lvert W\rvert$
	$K_iK_j(f_{ij}(w, \vec{a}_{-j}))$	$\lvert A\rvert^{\lvert A\rvert^n}\lvert W\rvert$	$\lvert R_j\rvert^{\lvert A\rvert^n}\lvert W\rvert$
	$K_iK_jK_k(f_{ijk}(w))$	$\lvert A\rvert^{\lvert W\rvert}$	$\lvert R_k\rvert^{\lvert A\rvert\lvert W\rvert}$

These results assume that the agent is trying to learn a fixed target function. But since the other agents are also learning, they are changing their behavior and, thereby the target function. In studying the effects of these *moving target* functions we found that the difference in the expected steady-state errors of two agents with different sample complexities reaches a peak for *slowly* moving target functions. This implies that the advantage of using simpler models is maximized in slightly dynamic MAS, not in stable or rapidly changing MASs. This result is not merely theoretical, but was previously observed in [2]. We hope to derive other similarly interesting results from our framework, and are working to make sure that these are useful, predictive theorems that can be used by agents designers, like ourselves. Our goal is an agent framework that can dynamically determine which models it should use/learn.

[1] R. Fagin, J. Y. Halpern, Y. Moses, and M. Vardi. *Reasoning About Knowledge.* MIT Press, 1995.

[2] J. M. Vidal and E. H. Durfee. The impact of nested agent models in an information economy. *ICMAS*, 1996. `http://ai.eecs.umich.edu/people/jmvidal/papers/amumdl/`.

This research has been funded in part by Digital Libraries Initiative under CERA IRI-9411287.