A Consistency Based Approach of Action Model Learning in a Community of Agents

(Extended Abstract)

C. Rodrigues, H. Soldano
LIPN, UMR-CNRS 7030
Université Paris 13, Sorbonne Paris Cité

G. Bourgne
LIP6, UMR-CNRS 7606
Université Pierre et Marie Curie, Sorbonne Universités

C. Rouveirol
LIPN, UMR-CNRS 7030
Université Paris 13, Sorbonne Paris Cité

1. INTRODUCTION

In this paper, we modelize a community of autonomous agents in which each agent acts in the environment following some relational action model [4], describing the effect to be expected when applying a given action in a given state. At some given moment, the underlying action model is only imperfectly known by agents and may have to be revised according to the unexpected effect of the current action. In a Multi Agent context, this revision process can and should benefit of interactions between the agents. For that purpose, we consider the general multi agent learning protocol SMILE [2] together with the relational action model learner IRALE [5] in order to modelize the interactions between agents.

2. SMILE

The SMILE protocol is based on a "consistency maintenance" process: after revising his current model in order to ensure that the revised model is *consistent* with the observations he has memorized, the agent communicates the revised model to the other members of the community, and possibly receives past observations they have memorized and that contradict the revised model.

Each agent r_i in a community of agents, or MAS, has at some moment a current belief set B_i, also denoted as his current model. He also has stored a set K_i of observations. A *consistency* property $Cons(B_i, K_i)$ is defined regarding beliefs and observations: r_i is said *a-consistent* iff $Cons(B_i, K_i)$ is true. Consistency of the agent with respect to the community is defined as the consistency of the agent with respect to the set of observations $K = K_1...K_n$ of all the agents of the MAS (including r_i): r_i is *mas-consistent* iff $Cons(B_i, K)$ is true.

When encountering a contradictory observation k making $Cons(B_i, K_i \cup k)$ false, an agent can apply a local revision mechanism M to recover his a-consistency. M changes B_i in B'_i and is such that observations coming from other agents may be used as if the agent had observed them. M and $Cons$ have then to satisfy the following properties:

- $Cons(B_i, K_i) \Rightarrow Cons(B'_i, K_i \cup k)$
- $Cons(B, K_1 \cup K_2)$ iff $Cons(B, K_1)$ and $Cons(B, K_2)$.

Appears in: *Alessio Lomuscio, Paul Scerri, Ana Bazzan, and Michael Huhns (eds.), Proceedings of the 13th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2014), May 5-9, 2014, Paris, France.*

Now, a global revision mechanism M_s allows the agent to restore its mas-consistency. The mechanism is triggered by an agent r_i upon direct observation of a contradictory observation k, denoted as an *internal counterexample*, thus enforcing revision of B_i into B'_i. To restore mas-consistency the agent starts a set of interactions with the other agents. Such an interaction $I(r_i, r_j)$ between the *learner* agent r_i and another agent r_j, acting as a *critic*, is as follows:

1. Agent r_i sends the revision B'_i to r_j.
2. Agent r_j checks the revision B'_i. If this modification preserves its a-consistency, r_j notifies r_i that he accepts B'_i, otherwise r_j sends to r_i a *contradictory* observation k', denoted as an *external counterexample*, such that $Cons(B'_i, k')$ is false.

The sequence of interactions ends when all critics accept the current revision B'_i.

What an agent makes of the mas-consistent models that other agents communicate to him leads to various instanciations of SMILE. In the *individualistic* variant iSMILE, an agent simply ignores other agent's models though he helps building them by communicating counterexamples [1].

3. IRALE

In IRALE, the agent knows which actions are available and has a complete representation of the current state, representing both his own state and the environment state. He sequentially performs actions that each possibly changes the current state into a new state, thus forming a *trajectory* in the space of states. The difference between these two states is considered as the *effect* of performing this action in the current state. IRALE describes states as a variable number of objects related by various relations, and represents the action model as a set B_i of STRIPS like first order rules.

Each rule r of B_i is composed of a precondition $r.p$, an action $r.a$ and an effect $r.e$. The precondition is a conjunction of positive literals, the action is a single literal, and the effect is made of two conjunctions of literals: $r.e.add$, to be added in the new state and $r.e.del$ to be withdrawn. The model has a default rule stating that whenever no rule applies, the action produces no effect, i.e. $e.del = e.add = \emptyset$. In the same way, each observation k is a state/action/effect triples $k = (k.s, k.a, k.e)$ observed during the agents history.

A rule r applies to an observation k, i.e. r *pre-matches* k, whenever its precondition/action parts matches the state/action parts of the observation, up to some grounding substitution of r. Applying r results in predicting the effect

$k.\hat{e}$. When it is the observed effect $k.e$, we say that r *post-matches* k. Observation k is contradictory when either there is no rule in B_i that predicts the observed effect or the predicted effect does not match the, possibly empty, observed effect.

The revision mechanism M involves both generalization and specialization operators [5]. Generalization, for instance, proceeds as follows: when no rule of B_i applies to k, a rule r of B_i is selected such that first, up to generalizing some constants into variables, r post-matches k, and, then, when applying *least general generalization* to r and k, the resulting rule pre-matches k and does not contradict past observations in K_i. Note that an IRALE agent only memorizes counterexamples, i.e. state/action/effect triples that have at some moment contradicted their current action model. Such memorization scheme ensures learning convergence in the realizable case to the target action model.

The IRALE agents experimented here are equipped with a symbolic planner and try to form and execute plans in order to reach individual goals. At some instant, the agent tries to build a plan. If planning succeeds, his current action model predicts the effect $\hat{e}$ in state s of the first action a of the plan, the agent performs then the action and observes the effect e. If $e = \hat{e}$, the agent applies the next action of the plan; otherwise, he memorizes a new counterexample and start a global revision by transmitting revised models to the other agents until all agents accept the current model. If planning fails, random actions are performed until an effect is observed, and planning is attempted again.

4. EXPERIMENTS

We consider a variant of the blocks world domain [5] in which color predicates are introduced: when the agent performs the action $move(a, b)$, a is actually moved on top of b only if a and b have the same color. Otherwise, a is not moved and its color shifts to the same color as b. When blocks are either back or white, the target action model needs 7 rules to model the action *move* and the state space is composed of nearly 5 million states.

Experiments are performed for communities of 1, 5 and 30 agents, each consisting in 100 runs. For each agent, a run is divided in episodes of at most 50 actions each. The agent starts the first episode with a null model and the current model at the end of an episode is the starting model at the beginning of the next episode. During an episode, the agent explores its environment, starting from a random state, and tries to reach a random goal, both provided by an external controller. Each agent uses FF [3] as a planner, and the planner is allowed five seconds to find a plan or state that planning has failed.

Experiments display a strong relation between the accuracy and the total number of revisions the agent has performed, i.e the number of internal and external counterexamples in his memory. When there are many agents such a memory size is obtained for far less actions than those performed by an isolated agent (see Table 1).

Figure 1 reports the average number of goals achieved by an agent during a run, as a task oriented measure of learning success. We observe that for all community sizes, there is a critical number of actions the agent has to perform before starting to produce accurate plans and to reach his random goals. This critical number is much smaller in the 30 agents community case but a clear benefit is yet obtained in the 5 agents community case.

Nb. ag.	Nb actions	Accuracy	Nb ex.	Nb intern. ex.
1	250	97.0	21.31	21.31
5	100	96.8	21.32	6.16
30	30	97.2	21.36	2.00

Table 1: Numbers of actions and counterexamples (total and internal) in an agent memory when accuracy reaches ≈ 0.97 in 1, 5 and 30 agents communities

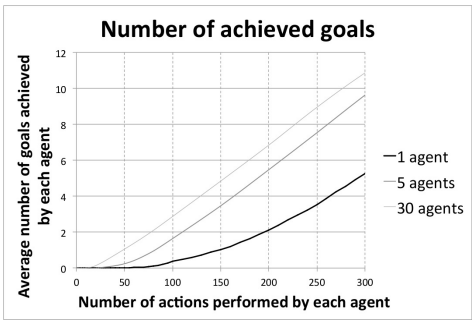

Figure 1: Number of goals achieved by an agent during a run in 1,5 and 30 agents communities

As a conclusion, each agent, when revising its current action model, benefits from past observations communicated by other agents on a utility basis: only observations contradicting the current model of the learner agent are transmitted. Agents are considered as autonomous entities, as for instance robots or mobile devices, with no access to controllers agents or shared memory. We argue that such autonomous entities, still able to communicate with similar entities while preserving their privacy, will play an important role in the future.

5. ACKNOWLEDGMENTS

The first integration of ISMILE and IRALE is due to Toru Mizuno as part of his master thesis work.

6. REFERENCES

[1] G. Bourgne, D. Bouthinon, A. El Fallah Seghrouchni, and H. Soldano. Collaborative concept learning: non individualistic vs individualistic agents. In *Proc. ICTAI*, pages 549–556, 2009.

[2] G. Bourgne, A. El Fallah-Seghrouchni, and H. Soldano. Smile: Sound multi-agent incremental learning. In *Proc. AAMAS*, page 38, 2007.

[3] J. Hoffmann. FF: The fast-forward planning system. *The AI Magazine*, 2001.

[4] T. Lang, M. Toussaint, and K. Kersting. Exploration in relational domains for model–based reinforcement learning. *Journal of Machine Learning Research (JMLR)*, 13:3725–2768, 2012.

[5] C. Rodrigues, P. Gérard, C. Rouveirol, and H. Soldano. Active learning of relational action models. In *Proc. Inductive Logic Programming (ILP) 2011*, volume 7207 of *LNCS*, pages 302–316. Springer, 2012.