Together, Is Anything Possible? A Look at Collective Commitments for Agents

Ben Wright

Department of Computer Science, New Mexico State University
Las Cruces, New Mexico, USA
bwright@cs.nmsu.edu

Abstract

In this research, commitments – specifically collective commitments – are looked at as a way to model connections between agents in groups. Using the concepts and ideas from action languages, we propose to model these commitments as actions along with the other basic actions that autonomous agents are capable of performing. The languages developed will be tested against different examples from various multi-agent system (MAS) areas and implemented to run in answer set programming.

1998 ACM Subject Classification I.2.4 Knowledge Representation Formalisms and Methods, I.2.11 Distributed Artificial Intelligence

Keywords and phrases Reasoning About Knowledge, Action Languages, Commitments, Multi-agent systems, Modal Logic

Digital Object Identifier 10.4230/LIPIcs.ICLP.2012.476

1 Introduction and problem description

Nowadays, it is becoming more frequent to see research mentioning 'distributed this' or 'that network'. Artificial Intelligence (AI) and knowledge representation are no different. For decades now, multi-agent systems and distributed AI have been vastly popular areas of study.

This project looks to study the organization and representation of *groups* - which we use loosely as a term. Borrowing from many well known research areas, we believe that a simple and declarative approach using ***action languages*** and ***commitments*** will be the key to modeling many of the concepts involved in modeling certain groups.

We wish to represent this idea in both a *declarative* and *logical* way. By declarative, we mean to say in the sense that the rules and constraints of the group will be represented formally. When we say in a logical way, we mean a way in which the rules and how the group may come to *reason* are done using aspects of traditional logics. This also comes into play when we wish to define things the group may do either in a temporal sense, for instance: "The group may do something in the future.", or in an epistemic way, like "The group already knew that it was done." These two areas have very active research areas in ***temporal*** and ***epistemic logic***.

In order to test our system, we will use answer set programming as it models *declarative* and *logical* concepts easily. Once finished, we believe this type of model will be able to represent features in group for different areas such as negotiations, normative systems, joint action plans, multi-agent planning, and argumentation to name a few.

Technical Communications of the 28th International Conference on Logic Programming (ICLP'12).
Editors: A. Dovier and V. Santos Costa; pp. 476–480

Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

2 Background and Overview of Existing Literature

2.1 Epistemic & Temporal Logics

Epistemic logic is focused on reasoning about the knowledge that agents may have about themselves, the world, or other agents. Two good references that define this area are [6] and [3]. Epistemic logic captures the essence of knowledge through modal operators. For instance $K_i p$ would read 'Agent i knows that the fluent p is true.' Operators are then defined on how knowledge may change or how knowledge can be *inferred* from other knowledge. Knowledge can also be nested, and this can be shown by a well known axiom rule called 'positive introspection' which is as follows: $K_i p \implies K_i K_i p$ which can be read as "If Agent i know p, then it knows that it knows p".

A big area of complexity and interest in this area is the idea of *common knowledge*. Common knowledge plays off of the idea "What does everyone know and knows that everyone knows?". From the previous example of positive introspection alone, we can see that this type of 'everyone knows' can become quite large in its representation.

Like epistemic logic, temporal logic is focusing on reasoning about truths that pertain to time. Many times in more 'real world' settings, fluents or values may change as the course of time or a program runs. Expressing these snippets of time may prove difficult if there isn't a mechanism to frame time in. There are two main ways to approach temporal logic: linear or branching. Linear temporal logic (LTL) treats time as a path along a string while branching temporal logic views time as a tree in which many different branches are possible, a well known system for this is Computation Tree Logic (CTL). There is actually a system known as CTL* which attempts to combine both of these ideas of representation together [5].

2.2 Action Languages

Representing what agents are capable of doing or how they may change their beliefs has always been a large area of research. One such idea for this representation is action languages. Action languages have been around for sometime now and have been shown as a way to set up *automated reasoning* or *planning* [9]. Various features have been implemented for agents using action languages like incomplete knowledge, reasoning about knowledge, and time.

Action languages work based on transitions from a state of fluents to another state of fluents. For instance, if "$move\_left$ **causes** $left$" is an action then when it occurs, $left$ would be true. Likewise, other features like static causal laws may show indirect effects like "$left$ **if** $\neg right$" which would mean that "$left$ is only true when $right$ is false" [7].

In addition *multi-agent* action languages (MALs) has been growing as well in which communication, coordination, and dispersion of fluentspaces/actionspaces all become issues of concern when developing. Some example MALs can be found in [12] and [1].

2.3 Commitments

Commitments have been studied for sometime in the MAS research area. Commitments formalize obligations or contracts between agents in a way that is logically representable. For instance if Agent A says that "I'll return the book to B within 2 days" then something like '$c(A, B, has\_book(B), 0, 2)$.' can be used to represent this.

By using this idea of commitments, more complicated ideas can be formalized. For instance, protocols for agent communication can be formed like "If someone sends me a message, I will respond with an acknowledgement". With protocols, concepts such as *negotiation* or *argumentation* can be reasoned about.

In addition to this *building up* idea, commitments can also represent the tone or 'mood' of a system. Much like in *normative systems* where norms or social choice assign 'basic behaviors' to all agents much like "All cars stop at a red light".

When commitments are moved beyond the scope of agent to agent (that is one of the sides is a group) there is some debate as to how these occur. These group, or *collective*, commitments become difficult to represent. What does it mean for "If the doorbell rings, someone in the door will answer it."? Will everyone in the house answer the door? Will everyone assume someone else answers the door? These are still open issues and there are some differences in how researchers approach them. For instance, some attach *intentionality* to commitments [4] while others do not [13].

3 Research Accomplishments, Goals, and Future

3.1 Goal

The goal of this research is to capture the behavior of groups effectively. To do this, we will use *collective commitments* and *action languages*. By using commitments and action languages, we will be able to represent the ideas in both a logical and declarative way that leads to simple transitions among state fluents.

An example of what we are trying to represent is the idea of having a group of merchants (suppose m_1, m_2, and m_3) and a group of consumers (suppose c_1, c_2, and c_3). With actions, we can define stuff similar to '*purchase_item* **causes** *item_bought*' for the consumers and '*offer_item* **causes** *item_known*'. Where *purchase_item* and *offer_item* are actions and *item_bought* and *item_known* are fluents for the consumers and merchants respectively.

If we now have a collective commitment by the merchants such as "If an item is offered, at least one of us has that item for sale" this can model an interesting behavior in a succinct way. Rather than stating "If m_1 offers an item, then either m_1 or m_2 or m_3 has said item in their store at the time of a purchase by a consumer" and then have similar rules for m_2 and m_3 as well.

In addition to this type of commitment, another form we are looking into are *epistemic commitments* which can represent things like "Consumers will only purchase items from merchants they believe are giving them a fair deal". In this example, 'fair deal' would have to be defined more explicitly. However the interesting parts of this commitment are twofold — first, it does not pertain to a specific agent, but a *group* of agents and second, it plays off of the *beliefs* that that agent holds.

By intertwining these areas together in a logical and declarative way with action languages and commitments, we feel that expressing the behaviors in groups of agents will be reduced greatly.

3.2 Current Status of the Research

The concept of collective commitments has already been approached in some research [4, 2]. However, these do not use the idea of *action languages*. [13] proposes an action language to model basic commitments, that is non-collective commitments. No implementation of the action language exists yet with commitments.

In addition to the concept of commitments, *beliefs* have also been researched for action languages. [8, 10, 11] all consider knowledge or belief in action languages. These however only focus on single agents. [1] proposes an action language that models Kripke Structures.

An implementation of *epistemic commitments* could not be found. Creating and implementation an action theory that allows for commitments and beliefs will constitute a large portion of this research.

Current progress on this research is attempting to implement base level commitment from action languages into an answer set program model. In use currently is the MAL $\mathcal{L}^{mt}$ defined in [13]. From this language we currently have delayed and reversible processes implemented. So ideas like "The payment should arrive in 3 to 5 days" and "I sent the payment 2 days ago, but wish to cancel it now" can be represented.

Connection these processes to simple commitments still needs to be done in addition to adding in belief systems. Following this, these ideas can be raised to the level of "groups".

3.3 Open Issues and Expected Achievements

Building up the system implementation to allow for more flexibility and group variance seems to exponentially grow the size of our stable models returned in ASP. As many of the smaller features in the logics and representation areas already belong in higher complexity classes, bringing it all together in a straightforward manner may prove difficult.

Also, consideration on the upper bound for the number of agents a system may have has not been considered in depth. With the aforementioned complexity issues, this will need to be considered, especially for the implementation part.

In addition, some of the features we wish to add —such as epistemic commitments— are still vague in what they actually mean or do. Some further research into specific case studies will need to be done.

At the end of this research, we expect to have a basic implementation of epistemic and group commitments using action theories and answer set programming. As these ideas bridge many different areas of MAS and Logic, we will use examples from these different areas to test our ideas and implementation. At this stage of research, the implementation will only be concerned with *satisfiability* of models, rather than *optimal* models.

References

1 Chitta Baral, Gregory Gelfond, Enrico Pontelli, and Tran Cao Son. Logic programming for finding models in the logics of knowledge and its applications: A case study. *TPLP*, 10(4-6):675–690, 2010.

2 Cristiano Castelfranchi. Commitments : From Individual Intentions to Groups and Organizations. In *Proceedings of the First International Conference on Multiagent Systems*, pages 41–48, 1995.

3 Hans van Ditmarsch, Wiebe van der Hoek, and Barteld Kooi. *Dynamic Epistemic Logic*. Springer Publishing Company, Incorporated, 1st edition, 2007.

4 Barbara Maria Dunin-Keplicz and Rineke Verbrugge. *Teamwork in Multi-Agent Systems: A Formal Approach*. Wiley Publishing, 1st edition, 2010.

5 E. Allen Emerson and Joseph Y. Halpern. "sometimes" and "not never" revisited: On branching versus linear time temporal logic. *J. ACM*, 33(1):151–178, January 1986.

6 Ronald Fagin, Joseph Y. Halpern, Yoram Moses, and Moshe Y. Vardi. *Reasoning About Knowledge*. MIT Press, 1995.

7 Michael Gelfond and Vladimir Lifschitz. Action languages. *Electronic Transactions on AI*, 3, 1998.

8 Aaron Hunter and James P Delgrande. An Action Description Language for Iterated Belief Change. In *IJAIC 2007*, pages 2498–2503, 2007.

9 Vladimir Lifschitz. Action languages, answer sets and planning. In *In The Logic Programming Paradigm: a 25-Year Perspective*, pages 357–373. Springer Verlag, 1999.

10 Jorge Lobo, Gisela Mendez, and Stuart R. Taylor. Adding Knowledge to the Action Description Language A. In *AAAI-97*, number 3, pages 454–459, 1997.

11 Marek Sergot and Robert Craven. The Deontic Component of Action Language n C +. In *DEON 2006*, pages 222–237, 2006.

12 Tran Cao Son, Enrico Pontelli, and Ngoc-Hieu Nguyen. Planning for multiagent using asp-prolog. In Jürgen Dix, Michael Fisher, and Peter Novák, editors, *Computational Logic in Multi-Agent Systems*, volume 6214 of *Lecture Notes in Computer Science*, pages 1–21. Springer Berlin / Heidelberg, 2010. 10.1007/978-3-642-16867-3_1.

13 Tran Cao Son, Enrico Pontelli, and Chiaki Sakama. Formalizing commitments using action languages. In Chiaki Sakama, Sebastian Sardiña, Wamberto Vasconcelos, and Michael Winikoff, editors, *DALT*, volume 7169 of *Lecture Notes in Computer Science*, pages 67–83. Springer, 2011.