Online appendix for the paper

Autonomous Agents Coordination: Action Languages meet CLP($\mathcal{FD}$) and Linda∗

published in Theory and Practice of Logic Programming

AGOSTINO DOVIER
Università di Udine, Dipartimento di Matematica e Informatica
(*e-mail:* `dovier@dimi.uniud.it`)

ANDREA FORMISANO
Università di Perugia, Dipartimento di Matematica e Informatica
(*e-mail:* `formis@dmi.unipg.it`)

ENRICO PONTELLI
New Mexico State University, Department of Computer Science
(*e-mail:* `epontell@cs.nmsu.edu`)

submitted 23 November 2010; revised 5 July 2011; accepted 31 August 2011

Appendix A The languages $\mathcal{B}^{\mathsf{AAC}}$, B^{MAP}, and B^{MV}

The language $\mathcal{B}^{\mathsf{AAC}}$, and its implementation, heavily relies on its foundations B^{MAP} and B^{MV}. In this section we briefly compare these three languages to clarify which parts of the solvers of the previous languages can be used for the implementation of $\mathcal{B}^{\mathsf{AAC}}$ presented in Subsection 3.6.

Let us focus first on B^{MV}. This is a single agent framework. Therefore, considering a given action theory, all fluents and actions are known to the single agent, and the language does not permit to specify private fluents or actions. Moreover, B^{MV} allows one to specify static causal laws. The syntax of fluent expressions and constraints is exactly the same as in $\mathcal{B}^{\mathsf{AAC}}$. The syntax for executability and action effects is analogous to that of $\mathcal{B}^{\mathsf{AAC}}$. More precisely, in B^{MV}, these laws take the forms:

- `exectuable`(a, C)
- `causes`(x, C_1, C_2), where C_1 is the constraint that will hold in the next state if the action x is executed in a state where C_2 holds.

These are just syntactical variants of (5) and (6), respectively. The semantics of B^{MV} is given via a transition system analogous to that introduced for $\mathcal{B}^{\mathsf{AAC}}$. In particular, one might note that if a $\mathcal{B}^{\mathsf{AAC}}$ action description involves a single agent that knows all

∗ Research partially funded by GNCS-INdAM projects, MUR-PRIN: *Innovative and multidisciplinary approaches for constraint and preference reasoning* project; NSF grants IIS-0812267 and HRD-0420407; and grants 2009.010.0336 and 2010.011.0403.

the fluents (and no communication laws are included), then its semantics coincides with the one of the corresponding B^{MV} program obtained by an immediat syntactical translation. The Prolog interpreter for B^{MV} is proved to be correct and complete (for soundness the absence of static laws is needed, but this is the case of $\mathcal{B}^{\mathsf{AAC}}$, as presented here) with respect to the semantics in (Dovier et al. 2010).

Let us consider now B^{MAP}. It is a multiagent, centralized language, where collective actions, namely actions that require more than one agent for being executed, are allowed. For instance, a law of the form

$$\texttt{action } x \texttt{ executable by } a_1, a_2, \ldots, a_n$$

specifies that agents $a_1, a_2, \ldots, a_n$ may execute together the action x. In $\mathcal{B}^{\mathsf{AAC}}$, instead, in the domain of an agent a, an action definition implicitly states that the action is executed by a (hence, this is a particular case of the B^{MAP} law). On the other hand, since the reasoner is centralized, conflicts among effects never occur and all (concomitant) planned actions are always successfully executed. The declaration of fluents in B^{MAP} is analogous to that in $\mathcal{B}^{\mathsf{AAC}}$, whereas B^{MAP} has a different syntax for dynamic laws, since they can refer directly to action-occurrences. A B^{MAP} dynamic law has the form *Prec* `causes` *Eff*, where *Prec* and *Eff* are constraints and at least one reference to an action x must explicitly occur in *Prec*. Such references are specified by exploiting action flags of the form `actocc(`x`)`.

The semantics of B^{MAP} is given via the same notion of transition system used for B^{MV} and for $\mathcal{B}^{\mathsf{AAC}}$. If a multi-agent action description in $\mathcal{B}^{\mathsf{AAC}}$, together with initial state and goal, is such that during the plan, no conflict occurs, then the B^{MAP} action description obtained by a simple (mostly one-to-one) translation, has exactly the same behaviour on the transition system. Let us observe that in this translation, collective actions are not generated.