Modelling Inertia in Action Languages
(Extended Report)

Mikhail Prokopenko¹ and Pavlos Peppas²

¹ CSIRO Mathematical and Information Sciences
Locked Bag 17, North Ryde, NSW 2113, AUSTRALIA
Email: mikhail.prokopenko@cmis.csiro.au
² Knowledge Systems Group
Department of Computing, School of MPCE
Macquarie University, NSW 2109, AUSTRALIA
Email: pavlos@mpce.mq.edu.au

Abstract. Logic-based approaches to reasoning about actions, change and causality, highlight efficient representation and processing of domain background knowledge as an important task. Action theories recently developed in the framework of action languages with inertia and ramifications [20,14] not only adopt the principle of minimal change reinforced with the policy of categorisation (assigning different degrees of inertia to language elements) but also try to incorporate background causal knowledge. In this paper we aim to trace the evolution of action languages and to explore interactions between ontological characteristics of action domains such as inertia and causality. Such an analysis should clarify how possible solutions to the frame and the ramification problems are affected by applying the policy of categorisation to causal domains. We first attempt to identify conditions (more precisely, restrictions) which preserve the meaning of domain descriptions when moving among various analysed languages. Relaxing such restrictions can help in evaluating the role of the frame concept (and policy of categorisation, in general) in an action language with fluent-triggered causality.

1 Introduction

Various investigations of the ramification problem have demonstrated that in order to deduce the indirect effects of actions it is necessary to efficiently represent background domain knowledge. The notion of efficiency is usually based on the idea of a compact and concise representation and dates back at least as far as Occam’s Razor: other things being equal, simple theories are to be preferred to complex ones. For instance, in the framework of first-order logic and its nonmonotonic extensions, representation of background knowledge in the form of domain constraints [6] is more economical than explicit description of actions consequences (such as in STRIPS [3]) because it avoids exponential difficulties usually associated with the latter¹. Appropriately represented knowledge at the

¹ It is not trivial to find a set of domain constraints always producing desirable inferences in the domain, and the complexity of this task sometimes is comparable
same time has to be efficiently "readable" and "processable" by an underlying reasoning mechanism. Such a reasoning system\(^2\) should be capable of inferring both direct and indirect consequences of a performed action (an occurred event) as well as preserve inertial properties of a domain (solve the frame problem [16]).

Action theories recently developed in the framework of action languages with inertia and ramifications [11,7,10,8] use the idea of minimising change to deduce the set of possible next states (successor states). The notion of minimal change is usually defined by set inclusion and incorporates the concept of frame, assigning different degrees of inertia to language elements (fluents, literals, formulas, etc.). For example, in [7] it is noted that "if F is not a frame fluent then it is not expected to keep its old value after performing an action, so that the change in its value is disregarded". In addition, some action theories try to represent domain knowledge in a way that not only provides a solution to the frame and the ramification problems but also embodies background information in the form of "causal laws" of a domain. Moreover, it was successfully argued in many publications [2,12,15,20] that, in general, propositions embracing causal dependencies are "more expressive than traditional state constraints" [20]. Following recent approaches to representing causal information, we do not intend here to investigate philosophical aspects of the phenomenon – probably discovering, understanding, and formalising an elusive nature of the cause - effect relation will by itself mark a significant improvement on the way to engineer artificial intelligence. However, we shall try to highlight possible areas of interaction between ontological characteristics of action domains such as inertia and causality and to address the following questions:

- under what conditions do we gain by dividing fluents on inertial and non-inertial ones in the framework of an action language with fluent-triggered causality\(^3\) (fact causality) and
- how does it affect possible solutions to the ramification problem.

We believe that the best way to proceed towards this goal is to trace the evolution of action languages.

### 2 Evolution of Action Languages

As mentioned in [14], the original idea behind introducing action languages was to present a methodology allowing for translation from a specialised action language to a general-purpose formalism, such as a nonmonotonic reasoning system based on first-order logic. A domain described in the first action language \(A\) [5], for instance, can be translated [9] into a logic programming language or into the circumscriptive approach of Baker [1]. Similarly in [11] there is a translation

---

2 The term "reasoning system" is used to refer to "any formal system that produces inferences about the effects of events" [17].

3 The terms action- and fluent-triggered causality are due to Lin [12].