Journal of Logical and Algebraic Methods in Programming 83 (2014) 319–338

Contents lists available at ScienceDirect

Journal of Logical and Algebraic Methods in Programming

www.elsevier.com/locate/jlamp

Foundations for structuring behavioural specifications

Răzvan Diaconescu [a,*], Ionuţ Ţuţu [b,c]

[a] Simion Stoilow Institute of Mathematics of the Romanian Academy, Romania
[b] Department of Computer Science, Royal Holloway University of London, United Kingdom
[c] Institute of Mathematics of the Romanian Academy, Research group of the project ID-3-0439, Romania

ARTICLE INFO

Article history:
Received 13 June 2013
Received in revised form 8 November 2013
Accepted 17 March 2014
Available online 13 April 2014

ABSTRACT

We develop foundations for structuring behavioural specifications based on the logic tradition of hidden algebra. This includes an analysis of a number of important technical compositional properties for behavioural signatures, such as pushouts, inclusions and unions, as well as an investigation of algebraic rules for behavioural module composition. As a particularity of behavioural specifications, some of the constructions and results arise in a partial algebraic form. This partiality aspect is one of the distinguishing features of our approach to behavioural specification modules. In addition, our study does not commit to any actual choice of structuring constructs, thus being applicable to a wide variety of structuring situations.

1. Introduction

Modern algebraic specification theory and practice has extended the traditional many-sorted algebra-based specification to several new paradigms. One of the most promising is behavioural specification, which originates from the work of Horst Reichel [31,32] and can be found in the literature under names such as hidden algebra [24,25], observational logic [4,27], coherent hidden algebra [19] and hidden logic [33]. Behavioural specification characterises how objects (and systems) *behave*, not how they are implemented. This new form of abstraction can be very powerful for the specification and verification of software systems since it naturally embeds other useful paradigms such as concurrency, object-orientation, constraints, nondeterminism, etc. (see [25] for details). In the tradition of algebraic specification, the behavioural abstraction is achieved by using specification with hidden sorts and a behavioural concept of satisfaction based on the idea of indistinguishability of states that are observationally the same, which also generalises process algebra and transition systems (see [25]).

An important effort has been undertaken to develop languages and systems supporting the behavioural extension of conventional or less conventional algebraic specification techniques; these include CafeOBJ [18,20], CIRC [35] and BOBJ [33]. In other situations, behavioural specification, although not directly realized at the level of the language definition, is employed as a mere methodological device [5]. In all cases there is the unavoidable need of a structuring mechanism for behavioural specifications. Structuring or modularization is in fact a common aspect of any formal method that aims at assisting the development of complex systems; without it such developments are simply not possible. Due to its important role in formal specification development (see for example [37]), the study of modularization is supported by a rather vast scientific literature developed over the past four decades. In the case of logic-based formal methods it has become standard to approach the study of modularization of systems through the so-called institution theory of Goguen and Burstall [23]. This trend has

* Corresponding author.
E-mail addresses: Razvan.Diaconescu@imar.ro (R. Diaconescu), ittutu@gmail.com (I. Ţuţu).

http://dx.doi.org/10.1016/j.jlamp.2014.03.001
2352-2208/

led to a fairly uniform understanding of various systems of modularization or structuring, one that has been successfully applied to the design of modern algebraic specification languages such as CASL [2] or CafeOBJ [18,20]. However, in spite of this, due to the inherent complexity and difficulty of the subject, the study of modularization is far from being closed as insufficiently explored aspects still exist. For instance, the recent works [17,15] attempt to give an answer as complete as possible to the issue of instantiating multiple parameters in a sharing context. Moreover, behavioural specification itself poses a particular challenge given by some of the specificities of its underlying logic that raise serious obstacles when attempting to apply established general theories on modularization. For example, one major source of problems is given by the fact that the union or aggregation of behavioural signatures is inherently partial rather than total and moreover, by noticing that this partiality is induced by two different factors. On the one hand, one may not aggregate signatures that share a sort name that is declared visible in one of the signatures and hidden in the other. On the other hand, any aggregation of signatures has to fulfil the encapsulation condition characteristic to behavioural signature morphisms, which cannot be guaranteed in all possible situations; this property is essential for the basic *satisfaction condition* of the underlying logic to hold, which in turn is absolutely necessary with respect to modularization. The requirement of both of these hypotheses (i.e. the preservation and the reflection of sorts' visibility, together with the encapsulation condition) has been extensively discussed in the literature (e.g. [25,27]), not only from a technical perspective, but also in terms of their practical relevance. Hence, under the current general assumptions on behavioural specification, the union of signatures arises naturally as a partial operation.

In this paper we develop foundations for structuring of behavioural specifications in the light of the most recent developments in structuring specifications in general [17,15]. This means reliance upon well established theoretical devices used in modularization studies such as institutions, pushouts, and inclusion systems. It also means that only some of the concepts and results already available at the general level may be applied directly, while much has to be reconsidered for the specific situation of behavioural specifications, thus leading to a series of new theoretical investigations.

The structure and the contents of the paper can be briefly described as follows. After a first preliminary section that recalls a number of concepts and results from previous works on the modularization of specifications on which we rely in our work, we present our contribution in two main sections:

1. The first one is devoted to our choice of a logical system that underlies behavioural specification. We introduce the institution of hidden algebra, detail the connection with related work, and develop a series of technical properties that are required by our modularization study. This includes pushouts and inclusion systems for behavioural signatures, as well as the investigation of several basic algebraic rules for the composition of behavioural signatures that are important for our study. A characteristic of these rules is that they are given in the style of partial algebras (in the sense of [9]). In contrast to their conventional corresponding variants they are conditional and, in addition, non-trivial to obtain. For example, the associativity of the union or aggregation of signatures, which in general follows immediately as a property of suprema, here, because of the partiality of the union of signatures, follows by reliance on a series of specific technical results.
2. In the second main section we introduce our basic framework for structuring behavioural specifications, which is based on recent ideas from [15]. An immediate consequence of the abstract nature of the central definitions is that we are able to develop our study on the structuring of behavioural specifications independently of any actual choice of specification building operators; this means direct applicability to a wide range of actual specification languages and systems. Moreover, the base institution is also considered abstractly by axiomatising some of the compositionality properties of hidden algebra; in this way, our work may be applied to other behavioural specification logics and even to other specification logics that share with hidden algebra some compositional properties.
   We also develop a few important algebraic rules for structured behavioural specifications, again in a partial algebra style that is inherited from the level of signatures. This partiality aspect is unique to our development since the module algebra literature [3,21,37] has considered thus far only total algebraic rules. In fact, our results constitute a generalisation of those module algebra works in the sense in which partial algebra is a generalisation of total algebra.

2. Preliminaries

In this section we recall a series of well-established concepts in the specification literature that are of central importance for the mathematical and logical foundations of modularization.

2.1. Categories

Institution theory relies technically upon category theory. We assume the reader is familiar with basic notions and standard notations from category theory. With few exceptions, in general we follow the terminology and the notations of [29]. With respect to notational conventions, $|\mathbb{C}|$ denotes the class of objects of a category $\mathbb{C}$, $\mathbb{C}(A, B)$ the set of arrows (morphisms) with domain A and codomain B, and ";" the composition (in diagrammatic order). A subcategory $\mathbb{C}'$ of $\mathbb{C}$ is