Dependent Types for an Adequate Programming of Algebra

Sergei D. Meshveliani *

Program Systems Institute of Russian Academy of sciences,
Pereslavl-Zalessky, Russia. http://botik.ru/PSI

Abstract. This research compares the author's experience in programming algebra in Haskell and in Agda (currently the former experience is large, and the latter is small). There are discussed certain hopes and doubts related to the dependently typed and verified programming of symbolic computation. This concerns the 1) author's experience history, 2) algebraic class hierarchy design, 3) proof cost overhead in evaluation and in coding, 4) other subjects. Various examples are considered.
Keywords: dependent types, computer algebra, functional language, `Agda, Haskell`.

1 Introduction

The author has a considerable experience in computer algebra, in provers based on term rewriting, and in programming this in `Haskell` [9], [10], [6]. But he is a newbie to the *dependently typed programming* [11], [1], [4], [8]. This paper contains the considerations and questions about the possibility of a workable computer algebra library based on the *dependently typed and verified* programming in `Agda`.

In 1995 – 2000 the author has been developing a computer algebra library `DoCon` [9], [10]. It is written in the `Haskell` language [6] and uses the tool of `Glasgow Haskell` [5]. The aim is to program algebra in a generic style, with defining the classical categories of `Group, Ring, Field`, and so on, and their instances for the classical *domain constructors*: `Integer, Fraction, Polynomial, ResidueRing`, and the such. The goal was to implement this approach to programming algebraic methods by using a *purely functional language*, having a data class system, and with making this library open-source.

1.1 Dynamic Parameter Domain

The most problematic point in the `DoCon` project is the subtle feature of modelling a *domain depending on a parameter*, especially when this parameter needs to be evaluated at the running time.

* This work is supported by the Program No.16 of Fundamental Research of the Russian Academy of Sciences Presidium ("Fundamentalnye Problemy Systemnogo Programmirovaniya").