FORMALIZING AN ABSTRACT ALGEBRA TEXTBOOK IN ISABELLE/HOL

J. ARANSAY AND J. DIVASÓN

Abstract. In this work we present the formalization of some well-known results in Abstract Algebra with the proof assistant Isabelle/HOL. Our interest focuses in finite-dimensional vector spaces, and our proposal is to closely follow the presentation made in a popular book of the field by P. R. Halmos. The main result in this work proves that every finite-dimensional vector space V over a field $\mathbb{K}$ is isomorphic to $\mathbb{K}^n$, with n the dimension of V. In the way to this result we had to make relevant choices on the representation of algebraic structures in the theorem prover, from which we inform in the paper. We also report on some informal "pen & pencil" proofs and arguments presented in Halmos, that we had to turn into algorithms for the proof assistant to accept them. The work fills a gap in our project of computing properties of generic finite-dimensional vector spaces by means of matricial algebra over fields, since it formally proves that both constructions are isomorphic.

Introduction

The Formath project [2] of the European Commission points out as one of its goals the formalization of the computation of an invariant of topological spaces known as homology groups. This involves computing the Smith Normal form of the incidence matrices of the topological space. The simplification of the original problem to the computation of the Smith Normal form relies on the possibility of representing finite-dimensional vector spaces over a given field $\mathbb{K}$ by means of their isomorphic counterparts, $\mathbb{K}^n$.

This result of Abstract Algebra has been used as a case study in the proof assistant Isabelle/HOL [5] to improve the knowledge of the system features to cope with finite-dimensional algebraic structures. At the same time, we wanted to learn about the difficulties that could emerge when a mathematical textbook is chosen to accurately follow its definitions, lemmas and proofs in a formalization project. In our work, the book chosen was an undergraduate textbook ("Finite-Dimensional Vector Spaces" by P. R. Halmos [4]). The reasons are that it almost does not assume previous facts and gives detailed proofs (from a traditional point of view) of every result.

It is generally assumed by mathematicians that mathematical texts (ranging from research papers to textbooks) are not always as precise (and sometimes even correct) as they should be (see for instance [3]), and thus in our work we had to cope with such inaccuracies.

The article is divided as follows: Section 1 introduces Isabelle, with a remark in the HOL-Algebra library. Section 2 presents our formalization of definitions, lemmas and proofs given in Halmos. Simultaneously, we comment on the differences between both versions.

This work has been supported by project MTM2009-13842-C02-01 (Ministerio de Educación y Ciencia) and by FORMATH project, nr. 243847, of the FET program within the FP7 of the European Commission.

1. Isabelle/HOL and algebraic structures

Isabelle is a generic theorem prover framework, since it permits the implementation of different logical systems on top of it. From these logical settings, the most popular one is Isabelle/HOL, which consists of an implementation of Higher-Order Logic (HOL). The HOL type system is rather simple, based on inductive data types, functional types (between already existing types), product types and type variables. Proofs are built by means of logical rules; statements must be expressions of *bool* type, and proofs are carried out by successive applications of rules or tactics, which are functions from type *bool* into *bool*.

Along the years Isabelle/HOL has evolved into a complex system including features which greatly improve the user experience. Some milestones in mathematics (the prime number theorem, the Fundamental Theorem of Algebra) have been reached, placing the feasibility of our project out of discussion. Nevertheless, the different proving efforts carried out in the system usually start from scratch and do not pay attention to code reusing. Even so, the system offers a library, being one of its parts devoted to Algebra. One of our aims in this work was to reuse previous definitions and results, that we briefly present in this section.

Algebraic structures in the HOL-Algebra library are implemented by means of record structures (record types are implemented as product types in which fields are labeled), where the domain of the algebraic structure and its operations are fields of the record. As an example, monoids are represented in the system in the following way:

```
record 'a monoid =
  carrier :: "'a set"
  mult    :: "['a, 'a] ⇒ 'a" (infixl "⊗₁" 70)
  one     :: 'a ("1₁")
```

The previous type represents every record with the aforementioned fields. Nevertheless, some requirements over its operations must be satisfied. These are introduced by means of a *locale*, a kind of module, which consists in a collection of conditions over terms of type monoid giving place to a *monoid* predicate:

```
locale monoid =
 fixes G (structure)
 assumes "⟦x ∈ carrier G; y ∈ carrier G⟧ ⟹ x ⊗ y ∈ carrier G"
 and "⟦x ∈ carrier G; y ∈ carrier G; z ∈ carrier G⟧ ⟹ (x ⊗ y) ⊗ z = x ⊗ (y ⊗ z)"
 and "1 ∈ carrier G" and "x ∈ carrier G ⟹ 1 ⊗ x = x" and "x ∈ carrier G ⟹ x ⊗ 1 = x"
```

The previous representation is not the only one in the system for algebraic structures. This one is particular in that it includes the *carrier* set of algebraic structures as an additional field, allowing us to represent special subsets (like the kernel of a morphism) in the same way algebraic structures are (just by replacing the carrier set by the corresponding one).

The HOL-Algebra library also provides definitions for abelian and multiplicative monoids, groups, and then structures such as rings and fields. Additionally, it contains proofs of properties over these structures and some additional tactics that automatize some proofs about addition, substraction or distributivity of elements. The library is not so rich in results relating these algebraic structures (for instance, it does not contain basic operations on morphisms between algebraic structures).

2. Implementation of Halmos' textbook in Isabelle

In this section we closely follow the first chapter of [4]. The book starts with a subsection where *fields* are introduced. Its Isabelle definition is already present in the HOL-Algebra library, based on a "record + locale" structure. We proved in Isabelle some basic properties, such as $(-\alpha)(-\beta) = \alpha\beta$. Proofs were exclusively based on equational reasoning.

Next subsection of Halmos presents the notion of *vector space*, not present in the library. We define a vector space V *over a given field* $\mathbb{K}$ by means of a *locale* which includes structures $\mathbb{K}$ and V, multiplication by scalars $\cdot_V$, and the assumptions of $\mathbb{K}$ being a field, V being an abelian group, and $\cdot_V$ satisfying $\alpha(\beta x) = (\alpha\beta)x$, $1x = x$, $\alpha(x + y) = \alpha x + \alpha y$ and $(\alpha + \beta)x = \alpha x + \beta x$, where $\alpha, \beta, 1 \in \mathbb{K}$ and $x, y \in V$ (no notion of finiteness has been introduced yet). In addition to providing a *locale* which represents vector spaces, we proved in Isabelle that any field $\mathbb{K}$ over itself is a vector space (with field multiplication as scalar operation); we also proved that for any natural n, $\mathbb{K}^n$ is a vector space over $\mathbb{K}$ (with pointwise multiplication as scalar product); we report later on the implementation of $\mathbb{K}^n$.

The next relevant subsection in Halmos presents the notion of linear dependence. Quoting Halmos, "a finite set $\{x_i\}$ of vectors is *linearly dependent* if there exists a corresponding set $\{\alpha_i\}$ of scalars, not all zero, such that $\sum_i \alpha_i \cdot x_i = 0$."; accordingly "if $\sum_i \alpha_i \cdot x_i = 0$ implies that $\alpha_i = 0$ for each i, the set $\{x_i\}$ is *linearly independent*". We must observe that $\{x_i\}$ is not simply a set, the author implicitly assumes the existence of an indexation among its elements. We preferred calling such structure an *indexed set* (or a *sequence*). Our Isabelle implementation of indexed sets $A = \{x_i\}_{i\in\{1\dots n\}}$ has been done by means of a cartesian product of the underlying set of elements and a bijection between $\{0 \dots (n-1)\}$ and $\{x_1 \dots x_n\}$. Accordingly, definitions for operations on sequences (insertion and removal) had to be defined. Additionally, we had to make use of libraries for *finite sums*.

In a somehow surprising way, next subsection of Halmos introduces the notion of *linear combination*. Prior to the notion of basis, a theorem claiming that "the set of non-zero vectors $x_1 \dots x_n$ is linearly dependent if and only if some x_k, $2 \leq k \leq n$, is a linear combination of the preceding ones" is proved; the result will be useful later. Halmos introduces in a new subsection the notions of *basis* and *finite-dimensional* vector spaces, as the ones containing a finite basis. The next interesting result claims that, in any finite-dimensional vector space V, "every linearly independent set can be extended to a basis". The argument of the proof is interesting; we consider any set $\{y_1, \dots, y_m\}$ of linear independent vectors and let $\{x_1 \dots x_n\}$ be a finite basis of V. We build the set $\{y_1 \dots y_m, x_1 \dots x_n\}$ (we can observe here that the indexing function of the sequence has to be modified accordingly). The *span* of this set is equal to V, and it is linearly dependent, so we can remove the *least* element being a linear combination of the preceding ones (this element must be proven to be between x_1 and x_n). We can either reach a linearly independent set or iteratively remove the following one. Being this process terminating, a linear independent set whose span is V (thus, a basis) is reached. The previous argument has an algorithmic nature that is implemented in Isabelle by means of a *tail recursive function*. This fact is indeed relevant, since the previous algorithm is not a total function (its result would be unexpected over ill-formed sequences), and the system has facilities that allow to introduce tail recursion under adequate termination conditions.

The next subsection proves that every two basis of a finite-dimensional vector space V have equal cardinality. The proof is based in a result which claims that for $\{x_1 \dots x_n\}$ any

linear independent set and $\{y_1 \dots y_m\}$ any generating set (a set whose span is equal to V), $n \leq m$. The proof is done by *reductio ad absurdum*: let us assume that $m < n$. A new set $\{x_1, y_1 \dots y_m\}$ is defined, which is generating and linearly dependent. Thus, some element y_i can be removed preserving the generating property. Iterating this process m times, a generating set $\{x_m \dots x_1\}$ is obtained; thus, any element $\{x_i |\, m+1 \leq i \leq n\}$ is combination of $\{x_1 \dots x_m\}$, but $\{x_1 \dots x_n\}$ was independent by hypothesis, so we reach a contradiction. Hence, $n \leq m$. Consequently, given X, Y any two basis, by symmetry, they contain the same number of elements. The previous artifact is implemented by means of a recursive function which is then applied exactly m times.

Next subsection of Halmos proves that "every n-dimensional vector space V over a field $\mathbb{K}$ is isomorphic to $\mathbb{K}^n$". The first concern is to produce a suitable representation of $\mathbb{K}^n$, being that Isabelle does not support dependent types. We came across the idea of defining elements of $\mathbb{K}^n$ as functions from *nat* to the underlying type of $\mathbb{K}$, that map every element $0 \leq i < n$ to elements of $\mathbb{K}$, and every element $n \leq i$ to $0_{\mathbb{K}}$. This last condition allows to compare elements of $\mathbb{K}^n$ through functions' extensional equality. In addition to this, we defined in Isabelle *linear transformations*, which are introduced in Halmos much later (even if they are implicitly used in this proof); we defined an isomorphism between the elements of the corresponding basis of V and $\mathbb{K}^n$, and extended it to the remaining elements by linearity.

Conclusions

In this paper we have tried to describe a *naive* attempt of formalization of a textbook in Abstract Algebra. Along the process of formalization we have had to face the lack of relevant details in definitions (as the one of sequences) or proofs (as the proof of V and $\mathbb{K}^n$ being isomorphic, which takes 5 lines in Halmos and *ca.* 3.500 in our Isabelle implementation, including previous definitions and facts). The effort has been worth, since we have produced interesting Isabelle libraries (a definition of finite vector spaces, a library for working with sequences or the definition of $\mathbb{K}^n$ for any field $\mathbb{K}$). We also have learned to implement some loose mathematical arguments (the ones in the proofs about linearly independent sets and the dimension of basis), in the form of recursive functions. These arguments pose a computational (or constructive) content, that has emerged explicitly in their formalization. Some of the definitions that we have formalized have been presented in other systems (for instance, in the proof assistant Mizar [6]), but they did not focus on finite-dimensional vector spaces and their dimension. The complete Isabelle development is available from [1].

References

[1] J. Divasón . Proofs of properties of finite-dimensional vector spaces using Isabelle/HOL. `http://www.unirioja.es/cu/jodivaso/degree_thesis/`
[2] Formath Project. `http://wiki.portal.chalmers.se/cse/pmwiki.php/ForMath`.
[3] T. Hales. Formal Proof. Notices of the American Mathematical Society, *55* (11). pp. 1370 – 1380. 2008.
[4] P. R. Halmos. Finite-Dimensional Vector Spaces. Springer, 1974.
[5] T. Nipkow *et al.* Isabelle/HOL: A proof assistant for Higher-Order Logic. Springer, 2002.
[6] W. A. Trybulec. Linear combination in Vector Spaces. Journal of Formalized Mathematics, *2*. 1990.

Departamento de Matemáticas y Computación, Universidad de La Rioja; Edificio Luis Vives, c. Luis de Ulloa – 26004. La Rioja. Spain.
E-mail address: `{jesus-maria.aransay},{jose.divasonm}@unirioja.es`