Abstract

Using the logical and formal framework of the *Theorema* project, we select a given representation of polynomials, which is defined formally in *Theorema* functor notation. This definition, taken from an interactive textbook about Groebner bases, makes use of 'sequence' variables, a special *Theorema* extension of the language of predicate logic. In order to contribute to the future goal of extracting the structure of an automatic prover from the structure of the corresponding domain definition, we support the creation of an automatic 'polynomial theorem prover' corresponding to the above representation. To specify how this prover will work we start developping the theory of polynomials.

For this we give manually composed example proofs about properties of the defined basic polynomial operators in *Mathematica* notebook format. We try to mimic the wanted output format, using the possibility to structure notebook proofs into subproofs, which can be studied at various levels of detail. We describe the methods used to create the proofs and discuss the similarities to human proving style and the advantages of computer support, on the levels of correctness and readability / understandability.

It should be noted that this is not a straightforward exercise, since we do not only present the proofs but also explain the necessary structure of the prover program, such as the use of two different induction principles or how to handle case distinction and rewriting.

In order to demonstrate the feasibility of programming such a prover we implement a simplified prototype and describe it by examining an example output of this prototype and comparing it with the corresponding manual proof.