A New Scheme for Computing with Algebraically Closed Fields

Allan Steel

School of Mathematics and Statistics
University of Sydney
NSW 2006 Australia
allan@maths.usyd.edu.au

Abstract. A new scheme is presented for computing with an algebraic closure of the rational field. It avoids factorization of polynomials over extension fields, but gives the illusion of a genuine field to the user. A technique of modular evaluation into a finite field ensures that a unique genuine field is simulated by the scheme and also provides fast optimizations for some critical operations. Fast modular matrix techniques are also used for several non-trivial operations. The scheme has been successfully implemented within the Magma Computer Algebra System.

1 Introduction

This paper presents a new scheme, implemented within the Magma Computer Algebra System [4], for computing with an algebraic closure of the rational field $\mathbf{Q}$. The scheme works by automatically constructing larger and larger algebraic extensions of $\mathbf{Q}$ as needed during a computation, thus giving the illusion to the user of computing with an algebraic closure of $\mathbf{Q}$. The defining polynomials are not necessarily irreducible over the subfields—factorization over algebraic number fields is avoided, and the defining polynomials are automatically modified when factors are found during computations with the field. These factors often arise naturally because of the structure of an algorithm which is computing over the field.

A similar scheme was already proposed before (the D5 system [6]), but in this case an algorithm based on the field must handle the parallelism which occurs when one must compute with several roots of a reducible polynomial, leading to situations where a certain expression evaluated at one root is invertible but evaluated at another root is *not* invertible.

The new scheme presented here has no such difficulty: all the roots of a squarefree polynomial are returned as distinct elements of a genuine field, and any algorithm working over a general field need not be modified in any way to handle the separate roots.

This paper concentrates on the theoretical model underlying the scheme; because of space restrictions, it is impossible to give detailed examples of how the scheme works in practice. For many examples and more information, see the chapter "Algebraically Closed Fields" in the Handbook of Magma Functions [3] or the same chapter in the Online Help of the Magma Homepage [10].

C. Fieker and D.R. Kohel (Eds.): ANTS 2002, LNCS 2369, pp. 491–505, 2002.

2 Definition of an Algebraically Closed Field

2.1 Basic Presentation

The main type of object which we will develop in this paper will be called an ACF, standing for "algebraically closed field". In the implementation, such an object will only be represented at any given moment by a certain finite-degree extension of $\mathbf{Q}$, but since the user will get the illusion that the field is algebraically closed, we will let ACF label the current object.

An ACF A will be represented by the quotient of a rank-n multivariate polynomial ring by a triangular ideal I with n defining polynomials (defined below). In general, I will not be a maximal ideal, so the quotient will not be a field. However, the other key component of A will be a certain sequence Γ of n elements in some finite field which will allow a "modular evaluation" technique, and this will have two separate but critical properties:

1. There will be a unique maximal ideal J containing I which is determined by I and Γ. Thus the quotient by J will define a unique field and the user will get the illusion of working with this field.
2. Some quick tests will be able to be performed in the finite field (via Γ), thus making some fundamental arithmetic operations very fast.

In this paper, we will always have the rational field $\mathbf{Q}$ as the base field, but the scheme can be made to work for any other base field for which one can implement a modular evaluation technique similar to the one described here.

2.2 Triangular Ideals

Throughout the paper, let $\mathbf{Q}_i$ denote $\mathbf{Q}[x_1, \ldots, x_i]$ and $\mathbf{Q}_0 = \mathbf{Q}$ and for $n \geq 1$, let $\mathbf{Q}_n$ have the lexicographical monomial ordering with $x_1 < x_2 < \cdots < x_n$ (see [5, Chap. 2, §2] for details on monomial orderings).

Definition 2.1. *A sequence of n polynomials $(f_1, \ldots, f_n) \in (\mathbf{Q}_n)^n$ is called a* triangular basis *if, for $1 \leq i \leq n$:*

1. *The greatest variable occurring in f_i is x_i.*
2. *f_i is monic, written as a polynomial in x_i.*

The trivial sequence $()$ is defined to be a triangular basis for $\mathbf{Q}_0 = \mathbf{Q}$. An ideal I of $\mathbf{Q}_n$ is called triangular *if it possesses a triangular basis (i.e., if it is generated as an ideal by some triangular basis).*

As an example, let $n = 3$ and $f_1 = x_1^2 + 1$, $f_2 = x_2^3 - x_1$ and $f_3 = x_3^2 - x_1 x_2 + 1$. Then (f_1, f_2, f_3) is a triangular basis in $\mathbf{Q}_3 = \mathbf{Q}[x_1, x_2, x_3]$.

It is easy to see that a triangular basis of an ideal I is a Gröbner basis of I (with respect to the above order). This means that we can form the unique normal form modulo I of an element in $\mathbf{Q}_n$. This is the only fact which we will use from the theory of Gröbner bases. See also [1,9] for more discussion concerning triangular ideals.