Radiroot

Roots Of A Polynomial By Radicals

A GAP4 Package

Version 1.0

by

Andreas Distler

Institut Computational Mathematics

Fachbereich Mathematik und Informatik

Technische Universität Braunschweig

Pockelsstr. 14

38106 Braunschweig

Germany

January 2006

Contents

1 Introduction

The main functionality of this package is to solve a rational polynomial by radicals and display the solution. That is possible iff the Galois group of the polynomial – a permutation group on its roots – is solvable. This fact has first been discovered by Évariste Galois (1811 – 1832), on whose ideas this implementation is based. The implemented algorithm is discribed in [Dis05].

For example, a radical expression for the roots of the polynomial $x^4 - x^3 - x^2 + x + 1$ is

$$\tfrac{1}{4} + \tfrac{1}{4}\sqrt{-3} + \tfrac{1}{2}\sqrt{\tfrac{7}{2} + \tfrac{1}{2}\sqrt{-3}}.$$

The package creates a LaTex file for the radical expression. Therefore you need a Latex compiler and the dvi viewer xdvi, to use the main functionality.

In addition to the readout you get several results in GAP. Some of them can be computed on their own. This are the splitting field of a rational polynomial and its Galois group as a permutation group on the roots.

This package uses the interface to KANT [DFK+97] in the package Alnuth to factorize polynomials over algebraic numberfields. This functionality must be available to use the functions in Radiroot.

2 Functionality of the Package

2.1 Methods for Rational Polynomials

1 ▶ `IsSolvable( f )`
▶ `IsSolvablePolynomial( f )`

returns `true` if the rational polynomial *f* has a solvable Galois group and `false` otherwise. It signals an error if there exists an irreducible factor with degree greater than 15.

For a rational polynomial *f*

2 ▶ `SplittingField( f )`

returns the smallest field, constructed with `FieldByPolynomial` (see Creation of number fields in Alnuth), that contains all roots of *f*.

```
gap> x := Indeterminate( Rationals, "x" );;
gap> f := UnivariatePolynomial( Rationals, [1,3,4,1] );
x^3+4*x^2+3*x+1
gap> L := SplittingField( f );
<field in characteristic 0>
gap> y := Indeterminate( L, "y" );;
gap> g := UnivariatePolynomial( L, One(L) * [1,3,4,1] );
y^3+!4*y^2+!3*y+!1
gap> Factors( g );
[ y+(-168/47-535/94*a-253/94*a^2-24/47*a^3-3/94*a^4),
  y+(336/47+488/47*a+253/47*a^2+48/47*a^3+3/47*a^4),
  y+(20/47-441/94*a-253/94*a^2-24/47*a^3-3/94*a^4) ]
gap> FactorsPolynomialKant( f, L );
[ y+(-168/47-535/94*a-253/94*a^2-24/47*a^3-3/94*a^4),
  y+(20/47-441/94*a-253/94*a^2-24/47*a^3-3/94*a^4),
  y+(336/47+488/47*a+253/47*a^2+48/47*a^3+3/47*a^4) ]
```

To factorize a polynomial over its splitting field one has to embed the polynomial first, as seen in the example, or use `FactorsPolynomialKant` (see Alnuth) instead of `Factors`. The primitive elment of the splitting field is always denoted by `a`.

3 ▶ `GaloisGroupOnRoots( f )`

calculates the Galois group G of the rational polynomial *f* as a permutation group with respect to the ordering of the roots of *f* given as matrices in *G!.roots*.

If you only want to get the Galois group itself it is often better to use the function `GaloisType` (see Chapter 64.11 in the GAP reference manual).

2.2 Solving a Polynomial by Radicals

1 ► `RootsOfPolynomialAsRadicals(` *f* `)`

computes a solution by radicals for the irreducible, rational polynomial *f* up to degree 15 if this is possible, i. e. if the Galoisgroup of *f* is solvable, and returns `fail` otherwise. The result is displayed in form of a dvi-file. Additionally a record *rec* is returned which contains the Galois group of *f* in the component `galgrp` and the splitting field. The Galois group is presented as permutation group on the roots which are available as list of matrices in *rec*`.galgrp!.roots`. The splitting field is given in two forms; on the one hand the matrix field `K` generated by the roots and on the other hand an algebraic number field `H` created by the defining polynomial of the matrix field. The component *rec*`.K!.cyclics` provides a list of pairs whose first entries are matrices which define the splitting field by gradual, cyclic extensions and whose second entries give the degree of the according extension. For the computation a root of unity that lies in the matrix field is used and can be found in *rec*`.K!.unity`

The computation may last very long and doesn't finish for every example if the degree of *f* is greater than 7.

2 ► `RootsOfPolynomialAsRadicalsNC(` *f*, *display* `)`

has the advantage that it can be used for polynomials with arbitrary degree. It does essentially the same as `RootsOfPolynomialAsRadicals` except that you can choose whether you want to create a dvi-file and display it or not by setting the boolean *display*. The function performs no test whether the polynomial *f* is irreducible. It also doesn't check at the beginning if *f* is solvable.

3 Installation

3.1 Getting and Installing this Package

This package is available at

```
http://www.icm.tu-bs.de/ag_algebra/software/distler/radiroot
```

in form of a gzipped tar-archive. For the installation instructions see Chapter 74.1 in the GAP Reference Manual. Normally you will unpack the archive in the 'pkg' directory of your GAP-Version by typing:

```
bash> tar xfz radiroot.tar.gz        # for the gzipped tar-archive
```

3.2 Loading the Package

To use the Radiroot Package you have to request it explicitly. This is done by calling

```
gap> LoadPackage("radiroot");
----------------------------------------------------------------------------
Loading  RadiRoot 1.0 (Roots of a Polynomial as Radicals)
by Andreas Distler (a.distler@tu-bs.de).
----------------------------------------------------------------------------
true
```

The `LoadPackage` command is described in Section 74.2.1 in the GAP Reference Manual.

If you want to load the Radiroot package by default, you can put the `LoadPackage` command into your `.gaprc` file (see Section 3.4 in the GAP Reference Manual).

3.3 Additional Requirements

The package Alnuth 2.0 or higher has to be available with its full functionality to use Radiroot.

A dvi file is created to display roots. As default the package uses the command `latex` searched for in your system programs to create the dvi file and the command `xdvi` to start the dvi viewer. If you can't use this settings you will have to change the function `RR_Display` in the file `Strings.gi` in the subdirectory `lib`.

Bibliography

[DFK+97] M. Daberkow, C. Fieker, J. Klüners, M. Pohst, K.Roegner, and K. Wildanger. Kant V4. *J. Symb. Comput.*, 24:267 – 283, 1997.

[Dis05] Andreas Distler. Ein Algorithmus zum Lösen einer Polynomgleichung durch Radikale. Diplomarbeit, TU Braunschweig, 2005.

http://www.icm.tu-bs.de/ag_algebra/software/distler/Diplom.pdf.

Index

This index covers only this manual. A page number in *italics* refers to a whole section which is devoted to the indexed subject. Keywords are sorted with case and spaces ignored, e.g., "PermutationCharacter" comes before "permutation group".