Computing subfields: Reverse of the primitive element problem

Daniel LAZARD, Annick VALIBOUZE[*][†]
LITP, Université P. et M. Curie,
4 place Jussieu,
F-75252 Paris Cedex 05

May 28, 1993

Abstract

We describe an algorithm which computes all subfields of an effectively given finite algebraic extension. Although the base field can be arbitrary, we focus our attention on the rationals.

This appears to be a fundamental tool for the simplification of algebraic numbers.

Introduction

Many algorithms in computer algebra contain subroutines which require to use algebraic numbers. Computing with them is especially important when polynomial systems of equation have to be solved. As an example let us consider the now called cyclic 7th–roots of unit, which are the solutions of the following system [4, 1]:

$$\begin{gathered}
a+b+c+d+e+f+g=0\\
ab+bc+cd+de+ef+fg+ga=0\\
abc+bcd+cde+def+efg+fga+gab=0\\
abcd+bcde+cdef+defg+efga+fgab+gabc=0\\
abcde+bcdef+cdefg+defga+efgab+fgabc+gabcd=0\\
abcdef+bcdefg+cdefga+defgab+efgabc+fgabcd+gabcde=0\\
abcdefg=1.
\end{gathered}$$

Some of the solutions of this system are of the form $(a,b,c,1/c,1/b,1/a,1)$ where a (and also b) is a root of following polynomial, and b and c are

*PRC–GDR Mathématique–Informatique
†e-mail: dl@litp.ibp.fr, avb@litp.ibp.fr

expressed as polynomials of degree at most 11 in a [6, 3, 10].

$$\begin{aligned} P(x) = \quad & x^{12} + 9x^{11} + 3x^{10} - 73x^9 - 177x^8 - 267x^7 - 315x^6 - 267x^5 \\ & -177x^4 - 73x^3 + 3x^2 + 9x + 1. \end{aligned}$$

Both from an intuitive and a computational point of view it is much more suitable to represent the root a of P by the "nested" system:

$$\begin{aligned} x^2 - 3x - 3 = 0, \\ y^3 + y^2 - 2y - 1 = 0, \\ a^2 - (xy - 1)a + 1 = 0. \end{aligned}$$

This much simpler representation has also the advantage that it gives some additional information on the Galois group of the polynomial P.

Generalizing this example, we can say that the simplification of algebraic numbers consists in two problems. The first one is to decompose the extension in which the algebraic numbers are expressed in subextensions of degree as low as possible; the second, is to choose generators for these subextensions which have a simple minimal polynomial and on which the algebraic numbers to be expressed have a simple form.

We consider here only the first problem, but we have to mention the function `polred` in the system PARI [2] which appears to be the best known solution for the second problem.

For solving the first problem, we give in this paper an algorithm for computing all subextensions of a given algebraic field extension $K \longrightarrow L$.

As a byproduct, the algorithm could also output the subfields of low degree and index of the extensions of low degree of L. However, we have to emphasize that the algorithm does not compute in general the Galois group nor the Galois closure, not even when the Galois closure may be obtained as a tower of extensions of low degrees.

We know of two previous algorithms ([5, 15], the latter seeming found by S. Landau) for finding the subfields of an extension. It is rather difficult to decide which of these three algorithms is the most efficient, because their most time consuming steps are factorization of very different shape. Moreover, for our algorithm, we need only the factors of low degrees as output of the factorization; thus the timings may very different when using specially designed factorization routines instead of the standard ones.

1 Symmetric resolvents

The notion of resolvent were introduced by Lagrange [9] for studying the solutions of an equation. We recall here the general definition of a resolvent which is needed for our algorithm. The idea is to transform an equation by means of an appropriate function in order to obtain some information about its roots. In this paper the appropriate functions will be symmetric polynomials.

Throughout the paper, p will be an irreducible univariate polynomial of degree n over a field K, which is generally the field of the rationals. The set of its roots in an algebraic closure of K will be denoted by $\alpha = \{\alpha_1, \ldots, \alpha_n\}$.

Let $X = \{x_1, \ldots, x_n\}$ be a set of n indeterminates. The symmetric group S_n acts naturally on $K(X)$ by permuting the elements of X. This action leads to the following definition of a resolvent. (The notation is the same as in [7].)

Definition 1 *Let H be a subgroup of S_n and let f be an element of $K(x_1, \ldots, x_n)$. The set of functions $f(x_{s(1)}, \ldots, x_{s(n)})$, for s in H, is called the* orbit *of f under the action of H and is denoted by $H(f)$.*

Definition 2 *Let f be an element of $K(x_1, \ldots, x_n)$ and p be a univariate polynomial, the roots of which are $\alpha_1, \ldots, \alpha_n$. The* (general) resolvent of p by f, *denoted by $f_*(p)$, is the polynomial whose roots are the elements of the orbit of f evaluated at the roots of p:*

$$f_*(p)(y) = \prod_{h \in S_n(f)} (y - h(\alpha_1, \ldots, \alpha_n)).$$

In this case f is called a transformation function.

The irreducible factors over K of a general resolvent are polynomials in y which are called *irreducible resolvents.*

When a function depends on exactly k variables we say that its *arity* is k. If a transformation function of arity k is symmetric on these k variables we say that the resolvent is *k-symmetric.* We will recall in Section 5 a method for computing resolvents, which is very simple in the case of symmetric ones. It is easy to see that when the transformation function is a polynomial the resolvent may also be obtained by using resultants (see [7, 9] and also [11], where Soicher uses them to compute linear resolvents.

2 Finding equations for subfields

In this section we give the main theorem of this paper. It gives a necessary condition on the symmetric resolvents, for the existence of a field between K and $K(a)$. This condition leads to our algorithm for computing all subfields of index k in $K(a)$.

In what follows $\mathrm{Irr}(b, K)$ will denote the minimal polynomial of the algebraic number b over the field K.

Theorem 1 *Let p be an irreducible polynomial over K of degree n and a be one of its roots. If L is a subfield of $K(a)$ of index k, then any k-symmetric resolvent $s_*(p)$ has a factor of degree n/k which is a power of an irreducible* *polynomial (over K) which has a root in L. More precisely, there exists a polynomial q and an irreducible polynomial h of degree d over K with a root in L such that $s_*(p) = qh^{n/(kd)}$*

Proof. Let $a = \alpha_1, \ldots, \alpha_k$ be the conjugates of a over L. The element $b := s(\alpha_1, \ldots, \alpha_k)$ is in L. This symmetric function can be written by means of the elementary symmetric functions of $\alpha_1, \ldots, \alpha_k$ which are, up to their sign, the coefficients of the minimal polynomial of a over L. Thus, $\mathrm{Irr}(b, K)$ has a degree d dividing $n/k = [L : K]$ and we have the following inclusion:

$$K \xrightarrow{d} K(b) \xrightarrow{n/(kd)} L \xrightarrow{k} K(a).$$

We have to prove that $\mathrm{Irr}(b, K)^{n/(kd)}$ divides $s_*(p)$, i.e., that b is a root of $s_*(p)$ of multiplicity at least $n/(kd)$. The symmetric group S_n acts on the sets of k conjugates of a, and thus acts also on the roots of $s_*(p)$. The multiplicity of b is $\mu := [\mathrm{Stab}_{S_n}(b) : \mathrm{Stab}_{S_n}(\{\alpha_1, \ldots, \alpha_k\})]$ (we recall that the roots of p are all distinct). On the other hand, let $H := \mathrm{Stab}_G(\{\alpha_1, \ldots, \alpha_k\})$ where G is the Galois groups of p over K; it is easy to verify that the map of homogeneous spaces $H/\mathrm{Stab}_G(b) \longrightarrow \mathrm{Stab}_{S_n}(\{\alpha_1, \ldots, \alpha_k\})/\mathrm{Stab}_{S_n}(b)$ is injective; this implies that μ is larger than the index of H in $\mathrm{Stab}_G(b)$. By Galois theory we know that $\mathrm{Stab}_G(b)$ and H are the automorphism groups of the splitting field of $K(b)$ and L; thus

$$[Stab_G(b) : H] = [L : K(b)] = n/(kd).$$

Hence we have $\mu \geq n/(kd)$. $\diamondsuit$

Remark 1 When looking for subfields of index k of $K(a)$, the following kinds of factorization may occur for the k-symmetric resolvent $s_*(p)$:

(a) $s_*(p)$ has a simple factor of degree n/k. Lemma 3 of Section 3 shows that this factor is the minimal polynomial of a generator of a subfield of index k of $K(a)$.

(b) $s_*(p)$ has an irreducible factor of degree n/k of multiplicity greater than 1. This factor defines an extension of K, but further computations are needed to decide whether this extension has a conjugate included in $K(a)$.

(c) $s_*(p)$ has a factor of degree n/k which is a power of an irreducible polynomial h of degree dividing n/k. This situation may correspond to a subfield of index k, which has itself a subfield with h as defining polynomial; further computations are needed for deciding if the subfield of index k exists and for finding it.

(d) $s_*(p)$ does not have a factor of degree n/k which is a power of an irreducible polynomial. This implies that $K(a)$ does not have a subfield of index k.

Our algorithm mainly consists in applying this remark to the elementary symmetric functions. We recall that the i-th *elementary symmetric function* of k variables, denoted $e_i(x_1, \ldots, x_k)$, is the sum of the elements of the orbit of the monomial $x_1 \ldots x_i$ under the action of the symmetric group S_n. It is also the coefficient of t^i in the polynomial $\prod_{j=1}^{n}(t + x_j)$.

We present now our algorithm for computing subfields ; beforehand, we introduce a data structure for representing the fields which appear as a tower of simple algebraic extensions embedded in $K(a)$:

Definition 3 *Let a be an algebraic number over K. We say that the list*

$$[\,[h_1, b_1, g_1], [h_2, b_2, g_2], \ldots, [h_r, b_r, g_r]\,],$$

is a descriptive chain of an intermediate field L between K and $K(a)$ if we have a sequence of field extensions :

$$K \xrightarrow{h_1} K(b_1) \xrightarrow{h_2} K(b_1, b_2) \xrightarrow{h_3} \cdots \xrightarrow{h_r} L = K(b_1, b_2, \ldots, b_r) \xrightarrow{g_r} K(a)$$

such that $h_i = Irr(b_i, K(b_1, \ldots, b_{i-1}))$ and $g_i = Irr(a, K(b_1, \ldots, b_i))$, for $i = 1, \ldots, r$.

The length *of this chain is r. The descriptive chain of K is the empty list* [ ].

Algorithm `Eqnfield(p, k)`

`Input :` `p` an irreducible univariate polynomial over K
`k` an integer dividing the degree of `p`
`a` a root of `p` appearing implicitly as the variable of `p`
`Output :` A list of descriptive chains representing all subfields of index `k` in $K($`a`$)$;

```
Begin
        Eqnfield2(p, [ ], k, 1)
End.
```

Algorithm `Eqnfield2(p, F, k, i)`

`Input :` `p` an irreducible polynomial over K
`a` a root of `p` appearing implicitly as the variable of `p`
`k` an integer dividing the degree of `p`
`F` a descriptive chain representing a field between K and $K($`a`$)$ of index greater than `k` in $K($`a`$)$
`i` the index of the symmetric function to use, which is also the depth of the recursion
`Output :` A list of descriptive chains representing all extensions of the field defined by `F`, of index `k` in $K($`a`$)$

```
Begin
   sol := empty
```

```
  n := deg(p)
  s := i-th elementary symmetric function over k variables
  if i > k then return empty                          {see Remark 2}
  R := s*(p)                                          {see Section 5}
  for all distinct irreducible factors h of R of degree dividing n/k
      d:=deg(h)
      m:=multiplicity(h,R)
      if d=1 and m≥n/k then                                {case (c)}
          sol := append(sol, Eqnfield2(p, F, k, i+1))
      elseif d=n/k and m=1 then                            {case (a)}
          b := a root of h in K(a), expressed as a polynomial in a
                                                      {see Section 4}
          sol := cons(endcons([h,b,Irr(a,F(b))], F), sol)
      elseif d=n/k and m>1 then                            {case (b)}
          for all root b of h such that b ∈ K(a) do
                                                {see Sections 3 and 4}
              sol := cons(endcons([h,b,Irr(a,F(b))], F), sol)
      elseif d<n/k and m≥n/(kd) then                       {case (c)}
          for all root b of h such that b ∈ K(a) do
                                                {see Sections 3 and 4}
              g := Irr(a,F(b))
              sol := append(sol,
                Eqnfield2(g, endcons([h, b, g], F), k, i+1))
      else             {h does not define a subextension of index k}
  return sol
End.
```

Remark 2 When a field of index greater than `k` is found, another symmetric functions is tryed, over the field generated by the preceding ones. The following lemma ensures that if a field of index `k` in K(`a`) exists, it is eventually found.

If one would try each new symmetric function with the same base field, instead of increasing it as in `Eqnfield`, the computations would probably be faster, but it may arise that none of the elementary symmetric function generates over K the subfield of index `k` to be computed.

Lemma 1 *If L is a subfield of index k in $K(a)$ and $a = \alpha_1, \ldots, \alpha_k$ are the conjugates of a over L, then L is generated over K by the k elementary symmetric functions in $\alpha_1, \ldots, \alpha_k$.*

Proof. The elementary symmetric functions of $a = \alpha_1, \ldots, \alpha_k$ are (up to sign) the coefficients of $\mathrm{Irr}(a, L)$. Thus they are in L. The field they generate is clearly of index k in $K(a)$, and is equal to L. $\diamondsuit$

3 Testing inclusion between fields

Assume that a k-symmetric resolvent $s_*(p)$ has an irreducible factor h of degree $m < n = \deg(p)$. By a permutation of the roots $\alpha = \{\alpha_1, \ldots, \alpha_n\}$ of p, we may choose a root $a = \alpha_1$ of p and a root b of h such that $b = s(\alpha_1, \ldots, \alpha_k)$.

We have the following diagram of field extensions

$$\begin{tikzcd}
 & K(a) \arrow[dr, "e"] & \\
K \arrow[ur, "n"] \arrow[dr, "m"'] & & K(a,b) \\
 & K(b) \arrow[ur, "r"'] &
\end{tikzcd}$$

where the labels of the arrows represent the degrees of the extensions.

Lemma 2 *Let s be a symmetric function of arity $k < n$ and h an irreducible factor of the resolvent $s_*(p)$. If h is a simple factor of $s_*(p)$, then the degree r of the extension $K(a,b)/K(b)$ is less than or equal to k.*

Proof. Let σ be a K–automorphism of the splitting field of $K(a,b)$ and $\alpha'_1, \ldots, \alpha'_k$ be the images of $a = \alpha_1, \ldots, \alpha_k$ by σ. If b is fixed by σ we have $s(\alpha'_1, \ldots, \alpha'_k) = b = s(\alpha_1, \ldots, \alpha_k)$; since b is a simple root of $s_*(p)$ we have $\{\alpha'_1, \ldots, \alpha'_k\} = \{\alpha_1, \ldots, \alpha_k\}$. Thus the orbit of a by the automorphisms of the splitting field of $K(a,b)$ which fix b may have at most k values ; since this orbit is the set of the conjugates of a over $K(b)$, the value k bounds the degree r of the extension $K(a,b)/K(b)$. $\diamondsuit$

Lemma 3 *Under the hypotheses of previous lemma, if the degree of the irreducible resolvent h is $m = n/k$, then $K(b)$ is a subfield of $K(a)$ and $r = k$.*

Proof. We have $ne = mr$, where e is the degree of the extension $K(a,b)/K(a)$. Previous lemma shows that $r \leq k$; thus $r \leq k = n/m = r/e$; hence $e = 1$ and $r = k$. $\diamondsuit$

Lemma 4 *Let p be an irreducible polynomial over $K[x]$ of degree n and a be a root of p. Let h be another irreducible polynomial over $K[x]$ of degree $m < n$.*

(i) If h has a linear factor $x - b$ in $K(a)[x]$ then $b \in K(a)$ and this factor gives an expression of b as a polynomial in a.

(ii) If p has an irreducible factor of degree n/m over $K[x]/(h)$ then h has a root b in $K(a)$ and the irreducible factor of p is $Irr(a, K(b))$, the minimal polynomial of a over $K(b)$.

Proof. Cases (i) and (ii) imply $e = 1$ in the previous diagram. $\diamondsuit$

Remark 3 In case (i) (resp. (ii)) we can obtain $Irr(a, K(b))$ (resp. $Irr(b, K(a))$) by using linear algebra. This will be discussed in next section.

Remark 4 Dixon [5] tests such an inclusion by means of a factorization of a polynomial of degree nm over Q.

4 Embedding subfields

At several steps of algorithm `Eqnfield`, we know of two extensions of K, given by the minimal polynomials (`p` and `h`) of some generators (`a` and `b`) ; we have seen in Section 3 how to test if the second extension may be embedded in the first one, and this gives the minimal polynomial $p_a = \mathrm{Irr}(a, K(b))$ or $p_b = \mathrm{Irr}(b, K(a))$.

$$\begin{tikzcd} & K(a) \arrow[dr, "p_b"] & \\ K \arrow[ur, "p"] \arrow[dr, "h"'] & & K(a,b) \\ & K(b) \arrow[ur, "p_a"'] & \end{tikzcd}$$

We show here how compute one of these minimal polynomials from the other. This can not be done by a second factorization, not only because of it cost, but also because we would have to choose the good factor among several ones.

Thus, we proceed by linear algebra, in the following way. As the problem is symmetric in a and b, we only need to describe how to compute p_a from p_b.

More precisely, we want a polynomial of degree r in a, with coefficients in $K(b)$; this means that these coefficients are univariate polynomials in b of degree strictly less than m. We take the coefficients of these polynomials in b as unknowns, i.e. we introduce mr new indeterminates u_{mi+j} and write :

$$p_a(x) = x^r + \sum_{i=0}^{r-1} U_i(b)x^i \quad \text{where} \quad U_i(b) = \sum_{j=0}^{m-1} u_{mi+j}b^j .$$

The polynomial p_a has to vanish for $x = a$. Computing $p_a(a)$ in $K(a)(b)$, i.e reducing the powers of b by p_b and the powers of a by p, we get a polynomial which is linear in the u_i and has a degree less than n in a and less than e in b. The fact that a is a root of p_a implies that the $en = mr$ coefficients of the monomials in a^ib^j are zero. This gives a square linear system ; a solution of it gives a value for the u_i and, thus, for the coefficients of p_a. Therefor, the linear system has a unique solution.

Remark 5 Suppose that we have $m < n$ and that we want to compute both p_a and p_b. We need one factorization over an algebraic extension. It is usually better to compute first p_a by factoring p over $K(b)$ and then p_b by linear algebra. In fact, the complexity of the available factorization routines depend more on the degree of the field extension than on the degree of the polynomial to factor. However, if p_b is factorized first, we only need the factors of degree 1, and this is probably faster if the factorization routine is customized in order to compute only these factors.

Remark 6 When p_a and p_b are known, it is easy to convert the representation of an element in $K(a)(b)$ (or in $K(a)$, if $e = 1$) to its representation in $K(b)(a)$ or conversely : it suffices to reduce by p_a and h or by p_b and p.

5 Computing the symmetric resolvents

The most natural way for computing general resolvents consists in using symmetric functions, but, for avoiding an exponential swell of the expressions, it is necessary to contract them and to keep only one term by orbit.

In this section, we describe briefly an algorithm for computing a k-symmetric resolvent of a univariate polynomial p of degree n. It appears in [14] and is improved for the particular case studyed here.

The degree of a k-symmetric resolvent is $\binom{n}{k}$; its coefficients may easily be deduced by Girard–Newton formulae from the *power functions* of the roots of the resolvent.

We recall that the i-th power function on a finite set A is the sum $\sum_{a \in A} a^i$. A *monomial form* on A is the sum of the orbit of a monomial under the action of S_n; more precisely, if I a k-uplet of positive integers (the exponent of the monomial) then $M_I(A) = \sum_{J \in S_n(I)} A^J$ is the corresponding monomial form. Generally we choose for I the *partition* (i.e., a decreasing sequence of integers) of the orbit. For example $M_{(2,1)}(x, y) = x^2y + xy^2$, the elementary symmetric functions, e_i are $M_{(1,\ldots,1)}$ and the i-th power function, p_i, is $M_{(i,0,\ldots)}$. Any symmetric polynomial is a linear combination of monomial forms, and any monomial form may be expressed as a polynomial in the power functions or in the elementary symmetric functions. For example $M_{(2,1)} = e_1e_2 - 3e_3 = p_2p_1 - p_3$.

With these notations we can describe the computation of the resolvent. Let $s(x_1, \ldots, x_k)$ be a symmetric function used as transformation function for computing the resolvent. Since s is symmetric, we express it on the base of monomial forms $M_I(x_1, \ldots, x_k)$. Let r be an integer between 1 and the degree $\binom{n}{k}$ of the resolvent ; using the product formula for symmetric functions [12], we can expand s^r on the same base, in order to obtain

$$s^r(x_1, \ldots, x_k) = \sum_{I \in E} c_I M_I(x_1, \ldots, x_k), \tag{1}$$

where E is some set of partitions, and the c_I are integers.

For computing the r-th power function of the roots of the resolvent, we have to compute the sum q of the elements of the orbit of $s^r(x_1, \ldots, x_k)$

under S_n (see definition 2). As s is symmetric, the expression of q is easily deduced from (1) :

$$q(X) = \sum_{I \in E} \binom{n - \lg(I)}{k - \lg(I)} c_I M_I(x_1, \ldots, x_k), \tag{2}$$

where $\lg(I)$ denotes the *length* of the partition I (the number of non-zero parts of I).

From this, the r-th power function of the roots of $s_*(p)$ is obtained by specializing the variables of q as the roots of p, i.e. by expressing q or the monomial forms appearing in it as polynomials in the symmetric functions of the roots of p, which are (up to the sign) its coefficients.

Remark 7 When the transformation function is the sum $x_1 + \cdots + x_k$, the polynomial q is easily obtained by the multinomial formula: in (2) the set E is the set of all partitions $I = (i_1, \ldots, i_k)$ of length at most k such that $i_1 + \cdots + i_k = r$ and the coefficient c_I of I is the multinomial coefficient $c_I = \binom{r}{i_1, \ldots, i_k}$. In this case one may also use the algorithm of Soicher [11] because s is linear.

Remark 8 Formula (2) avoids the computation of the action of the symmetric group. But an algorithm is needed for computing the expression of q in function of the coefficients of p without expanding monomial forms. Such an algorithm has been implemented in Macsyma [13].

6 An illustrating example

As an example, we show in this section how algorithm `Eqnfield` works on the polynomial P of degree 12 given in the introduction. We call a one of its roots.

This polynomial is clearly reciprocal and $d := a + 1/a$ generates a subfield of index 2. This element d is the sum of two conjugates of a ; thus its minimal polynomial may be obtained by factoring the resolvent by $x_1 + x_2$. But we already know a generator of the subfield ; thus its minimal polynomial may more easily be obtained by computing the minimal linear relation between the powers of d in $\mathbf{Q}(a)$. This minimal polynomial is

$$p(d) = d^6 + 9d^5 - 3d^4 - 118d^3 - 180d^2 - 3d + 43.$$

$\mathbf{Q}(a)$ has a subfield of index 3, not contained in $\mathbf{Q}(d)$, but, from now on, we restrict ourselves to the subfields of $\mathbf{Q}(d)$. Since the degree of a subfield divides the degree of the field, the only possible values for k are 2 and 3. For $k = 3$ and $s = x_1 + x_2 + x_3$ we have:

$$\begin{aligned} factor(s_*(p)) &= (x^2 + 9x + 15) \\ &\quad (x^6 + 27x^5 + 204x^4 + 27x^3 - 3681x^2 - 4698x + 13581) \end{aligned}$$

$$(x^6+27x^5+246x^4+720x^3-972x^2-5454x+2241)$$
$$(x^6+27x^5+246x^4+846x^3+729x^2-1107x-1161).$$

Thus, by case (a) of Remark 1, $\mathbf{Q}(d)$ has exactly one subfield of degree 2 generated by, say, $b_0 := (\sqrt{21}-9)/2$, such that $\mathrm{Irr}(b_0, \mathbf{Q}) = x^2+9x+15$. This subfield has a simpler generator $b := (3-\sqrt{21})/2 = -b_0-3$ with minimal polynomial $h_1 := \mathrm{Irr}(b, \mathbf{Q}) = x^2-3x-3$. This minimal polynomial has smaller coefficients ; we have preferred it on x^2-5x+1 (which generates the same field) because it gives simpler results in the computations which follow.

Similarly, for $k=2$ we obtain one subfield $F = \mathbf{Q}(c_0)$ of index 2 in $\mathbf{Q}(d)$, such that $\mathrm{Irr}(c_0, \mathbf{Q}) = x^3+9x^2+6x-43$; if we set $c_1 := c_0-3$ we obtain a new generator of F such that $\mathrm{Irr}(c_1, \mathbf{Q}) = x^3-21x-7$. A simpler generator is $c = (c_1-1)/3$ with minimal polynomial $h_2 := \mathrm{Irr}(c, \mathbf{Q}) = c^3+c^2-2c-1$; it is better because the powers of c generate the ring of the integers of F and also because this generator leads to a very simple final result. Note that $c = \omega + 1/\omega$ where ω is a primitive 7–th root of unity.

Now, as 2 and 3 are relatively prime, we have $\mathbf{Q} = \mathbf{Q}(b) \cap \mathbf{Q}(c)$; thus $\mathbf{Q}(b,c)$ is of degree 6 and equal to $\mathbf{Q}(d)$. We shall "compute" this equality by expressing d as a function of b and c, and also b and c as functions of d.

For this purpose we first compute the minimal polynomials of d over $\mathbf{Q}(b)$ and $\mathbf{Q}(c)$. This polynomials are respectively obtained by the factorizations :

$$\begin{aligned}
factor(p,h_1) &= (d^3+(b+3)d^2+(-4b-3)d-17b-14) \\
&\ (d^3+(-b+6)d^2+(4b-15)d+17b-65), \\
factor(p,h_2) &= (d^2+(-3c+2)d-3c^2-3c+1) \\
&\ (d^2+(-3c^2+8)d+3c-2) \\
&\ (d^2+(3c^2+3c-1)d+3c^2-8).
\end{aligned}$$

We find several polynomials of the same degree which correspond to the fact that $\mathbf{Q}(b)$ and $\mathbf{Q}(c)$ are Galois extensions and have several generators with the same minimal polynomial. In both cases we choose the first factor, which is the simplest ; let us call it h_3 and h_4, respectively.

This gives the following diagram of extensions, where the labels are the polynomials defining the extensions.

$$\begin{tikzcd}
 & \mathbf{Q}(b) \arrow[dr, "h_3"] & & \\
\mathbf{Q} \arrow[ur, "h_1"] \arrow[dr, "h_2"'] & & \mathbf{Q}(d) \arrow[r, "a^2-ad+1"] & \mathbf{Q}(a) \\
 & \mathbf{Q}(c) \arrow[ur, "h_4"'] & &
\end{tikzcd}$$

Now, having computed the minimal polynomial of each extension of this diagram, the algorithm of Section 4 may compute the expressions of b and c as polynomials of degree 5 in d or of degree 12 in a.

For computing the expression of d in term of b and c, we may reduce this expression of c by $h_3 = \mathrm{Irr}(d, \mathbf{Q}(b))$ and $h_1 = \mathrm{Irr}(b, \mathbf{Q})$, to obtain a polynomial c_{val} in b and d. Reducing similarly the monomials $b^i c_{val}^j$ for $i = 0, 1$ and $j = 0, 1, 2$, we get the expression in $\mathbf{Q}(d)$ of the elements of the canonical basis of $\mathbf{Q}(b, c)$. The expression of d as a linear combination of these basis elements may be found by solving a linear system, as in Section 4. This gives the expression $d = bc - 1$, which completes the simplification of a given in the introduction.

Remark 9 The only part of above computation which is not completely algorithmic is the choice of the generators of $\mathbf{Q}(b)$ and $\mathbf{Q}(c)$.

7 Using known facts

It arises frequently that some information on the subfields or on the Galois group is directly available or easily detected. This is the case (as above) for reciprocal polynomials and for decomposable polynomials of the form $h(g(x))$, for example the even polynomials of the form $p(x^2)$.

Incidentally, the present paper may be viewed as a generalization of decomposition algorithms (see [8]) not only because any decomposition defines a subfield, but also because we compute a decomposition of the form $h(g(x)) \equiv 0 \ mod \ p(x)$.

For all of above cases we know a generator of a subfield, $a + 1/a$, $g(a)$ or a^2 , where a is a root of the initial polynomial. When a generator of a subfield is known, there is no need to compute a resolvent nor to factorize : the minimal polynomial of the generator of the initial field over the intermediate one may be computed by linear algebra, with the method of Section 4.

Sometimes, an automorphism of the field $\mathbf{Q}(a)$ is known. This is the case for reciprocal and even polynomials. When this arise, a symmetric function of the images of the field generator by the powers of this automorphism gives a generator of a subfield, the index of which is generally the order of the automorphism.

For example, in the case of a reciprocal polynomial, the automorphism is $x \rightarrow 1/x$, and $a + 1/a$ is the result of the symmetric function $x_1 + x_2$ applied to a and its image $1/a$; in this case, the symmetric function $x_1 x_2$ gives 1 which generates the trivial extension.

With even polynomials, the automorphism is $x \rightarrow -x$. Taking the sum as symmetric function leads to a trivial result, but the symmetric function $-x_1 x_2$ gives a^2 which is usually taken for simplifying even polynomials.

For the above polynomial of degree 12, an automorphism of order 3 was known[3], which consists in replacing a by the second component b of the solution of cyclic 7th roots problem. The decomposition described in the introduction were first found by using this automorphism, without computing resolvents.

This possibility of using known automorphisms or known generators of subfields is in practice very important, because it avoids the computation of the resolvent and any factorization, which may be rather time consuming.

References

[1] Baeckelin, J. and Fröberg, R (1991). How we proved that there are exactly 924 cyclic 7–roots.*Proceedings of ISSAC'91* (S.W. Watt ed.). ACM Press (New–York), 103–111.

[2] Batu, C., Bernadi, D., Cohen, H. and Olivier, M. (1991). *User's Guide to PARI-GP*. Available by anonymous ftp from math.ucla.edu (128.97.4.254).

[3] Björck G. and Fröberg, R., (1989), A faster way to count the solutions of inhomogeneous systems of algebraic equations, with applications to cyclic n-roots, *Reports, Matematiska Institutionen, Stockholms Universitet, 1989-No 7.*

[4] Davenport, J.H. (1987). Looking at a set of equations. *Technical report 87–06*, University of Bath.

[5] Dixon, J.D. (1990). Computing subfields in algebraic number fields. *J. Austral. Math. Soc. (serie A)* **49**, 434-448.

[6] Faugère, J.C., Gianni, P., Lazard, D. and Mora, T. (1989). Efficient Computation of Zero–dimensional Gröbner Bases by Change of Ordering. *Submitted to J. Symb. Comp.* Technical Report LITP 89–52.

[7] Giusti, M., Lazard, D. and Valibouze, A. (1988). Algebraic transformations of polynomial equations, symmetric polynomials and elimination. *Symbolic and Algebraic Computation, International Symposium ISSAC '88* (P. Gianni, ed.), Lect. Notes in Comp. Sc. **358**, 309–314

[8] Kozen, D and Landau, S (1989). Polynomial decomposition algorithms. *J. Symb. Comp.* **7**, 445–456.

[9] Lagrange, J.L., (1770–1771). Réflexions sur la résolution algébrique des équations. *Nouveaux Mémoires de l'Académie royale des Sciences et Belles-Lettres de Berlin.*

[10] Lazard, D. (1992). Solving zero–dimensional algebraic systems. *J. Symbolic Computation* **13**, 117–131.

[11] Soicher, L. (1981).*The computation of the Galois groups.*. Thesis, Concordia University, Montreal (Quebec, Canada, 1981).

[12] Valibouze, A. (1987). Fonctions symétriques et changements de bases. *European Conference on Computer Algebra, Leipzig, GDR, 1987.* (J.H. Davenport, ed.) Lect. Notes in Comp. Sc. **378**.

[13] Valibouze, A. (1989). Symbolic computation with symmetric polynomials, an extension to Macsyma. *Computers and Mathematics (1989, MIT, Cambridge, Mass.)*. Springer-Verlag.

[14] Valibouze, A. (1989). Résolvantes et fonctions symétriques. *Proc. of the ACM-SIGSAM 1989 Intern. Symp. on Symbolic and Algebraic Computation, ISSAC'89 (Portland, Oregon).* ACM Press, 390-399.

[15] Zippel , Book to appear.