Testing Nilpotence of Galois Groups in Polynomial Time [1].

V. ARVIND, Institute of Mathematical Sciences,
CIT Campus, Chennai, India 600113.
arvind@imsc.res.in.
PIYUSH P KURUR[2], Department of Computer Science and Engineering,
Indian Institute of Technology, Kanpur,
Kanpur, UP 208016, India.
ppk@cse.iitk.ac.in

We give the first polynomial-time algorithm for checking whether the Galois group $\mathrm{Gal}(f)$ of an input polynomial $f(X) \in \mathbb{Q}[X]$ is nilpotent: the running time of our algorithm is bounded by a polynomial in the size of the coefficients of f and the degree of f. Additionally, we give a deterministic polynomial-time algorithm that, when given as input a polynomial $f(X) \in \mathbb{Q}[X]$ with nilpotent Galois group, computes for each prime factor p of $\#\mathrm{Gal}(f)$, a polynomial $g_p(X) \in \mathbb{Q}[X]$ whose Galois group of is the p-Sylow subgroup of $\mathrm{Gal}(f)$.

Categories and Subject Descriptors: F.2.1 [**Numerical Algorithms and Problems**]: Computations on polynomials

General Terms: Algorithms, Theory

Additional Key Words and Phrases: Galois group, nilpotence, univariate polynomials, polynomial time

1. INTRODUCTION

Computing the Galois group of a polynomial $f(X)$ with rational coefficients is a fundamental problem in algorithmic number theory. For a polynomial $f(X)$ it is well known that the Galois group $\mathrm{Gal}(f)$ acts faithfully as a permutation group on the roots of $f(X)$. Thus, for a polynomial f without repeated factors, we can think of the Galois group of f as a group of permutations on the set of distinct roots of f. In this paper, by computing the Galois group, we mean finding a generating set for the Galois group as a permutation group. This characterizes the group only up to a relabeling of the roots. However, this gives $\mathrm{Gal}(f)$ a compact representation because any subgroup of S_n has a generating set of size at most $n-1$ [Jerrum 1986]. Furthermore, since there is a substantial library of efficient algorithms for permutation groups that takes as input subgroups of S_n given by generating sets, this compact representation of Galois groups can be computationally useful. The book by Seress [Ákos Seress 2003] contains a comprehensive treatment of permutation group algorithms.

There are algorithms for computing the Galois group of polynomials over rationals that even go back to the nineteenth century [Tschebotaröw and Schwerdtfeger 1950]. However, no general polynomial-time algorithm for this problem is known to date. Asymptotically, the best known algorithm is due to Landau [Landau 1984]: given a polynomial $f(X)$, as a list of its coefficients in binary, it takes time polynomial in its input size and the order of Galois group of f. Landau's algorithm explicitly lists all elements of its Galois group $\mathrm{Gal}(f)$. However, for a degree n polynomial $f(X)$, $\mathrm{Gal}(f)$ can have $n!$ many elements. Hence, Landau's algorithm takes exponential time in the worst case. It is a long standing open problem if there is an asymptotically faster algorithm. Lenstra's survey [Lenstra Jr. 1992] discusses this and related problems.

[1] preliminary version of this paper was presented at the MFCS 2006 conference [Arvind and Kurur 2006](see also [Kurur 2006])

[2] Most of the work was done as a Ph.D student at the Institute of Mathematical Sciences.

In the absence of efficient asymptotic algorithms, considerable research has gone into designing practical algorithms for Galois group computation. The work of Stauduhar [Stauduhar 1973] describes an algorithm that could compute the Galois group of polynomials up to degree 8. Recent implementations with the computer algebra software KANT apparently work up to degree 15. This method is described in detail by Cohen [Cohen 1993]. Stauduhar's method has the drawback of precision problems as it involves numerical approximation of the roots of the polynomial. Instead of the numerical approximation, using p-adic approximation Geissler and Klüners [Geissler and Klüners 2000] gave a variant of Stauduhar's algorithm that could work for polynomials up to degree 15. Another well-studied practical method [Soicher and McKay 1985; Mattman and McKay 1985], which avoids the precision problems due to root approximations, works by computing certain invariants called the absolute resolvents of the given polynomial. However, this approach has the drawback of being computationally more intensive as it involves factoring of very large degree resolvent polynomials. An entire special issue [Matzat et al. 2000] is devoted to algorithmic Galois theory, with practical implementations as the main goal.

Although computing the Galois group of a polynomial remains hard in general, often it is sufficient to check whether the Galois group satisfies some specific property. Knowing the nature of the Galois group of a polynomial can provide insight into the structure of the roots of the polynomial. There is no better example than the celebrated work of Galois [Galois 1830a; 1830b] in which he shows that a polynomial $f(X)$ over rationals is solvable by radicals if and only if its Galois group is solvable.

In this paper we study the problem of testing nilpotence of the Galois group of a polynomial $f(X)$ over $\mathbb{Q}$. Landau's algorithm [Landau 1984] to compute the Galois group, although exponential in general, yields efficient algorithms in certain cases. For example, we can test whether the Galois group is abelian [Landau 1984] in polynomial time. We give a quick overview. Assume that the input polynomial $f(X)$ is irreducible; otherwise factor $f(X)$ by applying the LLL algorithm [Lenstra et al. 1982] and test whether the Galois group of each of its irreducible factors is abelian. Since the Galois group of $f(X)$ is a subgroup of the direct product of the Galois groups of each of its irreducible factors, this is clearly sufficient. Any abelian transitive group of S_n is of order n. Hence, if the input polynomial is irreducible and abelian then its Galois group is polynomially bounded. One can use Landau's algorithm [Landau 1984] and compute the Galois group explicitly and verify that it is abelian. Similarly, for solvability and nilpotence test, we can assume that the input polynomial is irreducible. However, transitive solvable subgroups of S_n can be of size exponential in n. As far as efficient algorithms are concerned, this does not give a satisfactory answer to the problem of checking whether a given polynomial is solvable by radicals. Landau and Miller made a remarkable breakthrough by giving a polynomial-time algorithm for checking whether the Galois group of an input polynomial is solvable[Landau and Miller 1985].

The class of nilpotent groups is a subclass of the class of solvable groups that contains the class of abelian groups. Just as in the case of solvable groups, transitive nilpotent subgroups of S_n can have order exponential in n. This rules out a nilpotence test which explicitly computes the Galois group. Besides, the Landau-Miller solvability test does not give a nilpotence test. A key idea used in the Landau-Miller algorithm is to reduce the problem of checking the solvability of the Galois group G of the input polynomial into checking the the solvability of each factor group G_{i-1}/G_i in some special composition series $G = G_0 \rhd \ldots \rhd G_n = 1$ of G. Despite each of the groups G_i being large, Landau-Miller could compute enough information of the factor groups G_{i-1}/G_i implicitly from the input polynomial. However, one cannot infer nilpotence of G, even if all the factor groups G_{i-1}/G_i are given explicitly. The simplest example that illustrates this are the groups S_3 and $\mathbb{Z}_2 \times \mathbb{Z}_3$. The factor groups are $\mathbb{Z}_2$ and $\mathbb{Z}_3$ in the composition series for both the groups. However, S_3 is not nilpotent (though it is solvable) and $\mathbb{Z}_2 \times \mathbb{Z}_3$ is nilpotent (even cyclic). Thus, the composition factors a group alone do not suffice to determine if it is solvable or nilpotent or even abelian.

Overview of our result

We give the first deterministic polynomial-time algorithm for testing whether the Galois group of an input polynomial $f(X) \in \mathbb{Q}[X]$ is nilpotent. The running time of our algorithm is bounded by a polynomial in $\operatorname{size}(f)$ and thus is polynomial in the input size. Although our algorithm is polynomial time, it is unlikely to perform well in practise as it involves factoring polynomials over number fields. Testing nilpotence has been addressed before from the point of view of developing practical algorithms. For example, Fernandez-Ferreiros and Molleda [Fernandez-Ferreiros and Gomez-Molleda 2003] have given an algorithm for testing nilpotence by computing the centre of the Galois group of f. A key step in their algorithm, based on the Chebotarev density theorem, is to pick primes whose Frobenius give elements in the centre of $\operatorname{Gal}(f)$. However, the worst case running time of this algorithm is polynomial in $\operatorname{size}(f)$ and the order $\#\operatorname{Gal}(f)$ of the Galois group which is exponential in the input size.

We now give a brief overview of the main idea behind our algorithm. The main observations that lead to the polynomial-time algorithm are Theorems 5.9 and 5.10, which together give a characterisation of transitive nilpotent permutation groups in terms of its block structure. Explicitly testing for this characterisation will require us to compute the Galois group and hence is infeasible. As in the Landau-Miller solvability test, however, we test this implicitly.

The normality of Sylow subgroups of nilpotent groups plays an important role in the proof of Theorems 5.9 and 5.10. As a byproduct of our main result we obtain the following additional polynomial-time algorithm: given a polynomial $f(X) \in \mathbb{Q}[X]$ with nilpotent Galois group, for each prime factor p of $\#\operatorname{Gal}(f)$ we can efficiently compute a polynomial $g_p(X)$ such that the Galois group $\operatorname{Gal}(g_p(X))$ is the p-Sylow subgroup of $\operatorname{Gal}(f)$.

2. GALOIS THEORY OVERVIEW

In this section we recall some basic Galois theory. A detailed presentation is available in a standard algebra textbook, like Lang's book [Lang 1999]. Let L and K be fields. We say that L is an *extension* of K and denote it by L/K if $L \supseteq K$. For an extension L/K, L is a vector space over K and by the *degree* of L/K, denoted by $[L : K]$, we mean its dimension. An extension L/K is *finite* if its degree $[L : K]$ is finite. If L/M and M/K are finite extensions then $[L : K] = [L : M].[M : K]$. The ring of polynomials over K with indeterminate X will be denoted by $K[X]$. This ring is a *unique factorisation domain.*

An element α in an extension L of K is *algebraic* if there is a polynomial $f(X)$ over K such that $f(\alpha) = 0$. For such an α the *minimal polynomial* over K is the unique monic polynomial $\mu_\alpha[K](X)$ over K of least degree for which α is a root. We write $\mu_\alpha(X)$ for $\mu_\alpha[K](X)$ when K is understood. Elements α and β in L are *conjugates* over K if they have the same minimal polynomial over K. The smallest subfield of L containing K and α is denoted by $K(\alpha)$. If α is algebraic over K then the field $K(\alpha)$ is isomorphic to the quotient $K[X]/\mu_\alpha[X]$.

Let $f(X)$ be a polynomial over K. By the *splitting field* of f over K, denoted by K_f we mean the smallest extension of K containing all the roots of f. A finite extension L/K is *normal* if for all irreducible polynomials $f(X)$ over K, either $f(X)$ splits or has no root in L. Any finite normal extension over K is the splitting field of some polynomial in $K[X]$. An extension L/K is *separable* if for all irreducible polynomials $f(X) \in K[X]$ there are no multiple roots in L. A normal and separable finite extension L/K is a *Galois extension.*

The *Galois group* of L/K, denoted by $\operatorname{Gal}(L/K)$, is the subgroup of automorphisms σ of L that leaves K fixed, i.e. $\sigma(\alpha) = \alpha$ for all $\alpha \in K$. The Galois group of a polynomial $f(X)$ over K, denoted by $\operatorname{Gal}(f)$, is the Galois group $\operatorname{Gal}(K_f/K)$ of its splitting field. For a subgroup G of automorphisms of L, the *fixed field* L^G is the largest subfield of L fixed by G. We now state the fundamental theorem of Galois theory [Lang 1999, Theorem 1.1, Chapter VI].

THEOREM 2.1 (FUNDAMENTAL THEOREM OF GALOIS THEORY). *Let L/K be a Galois extension with Galois group G. Let $\mathcal{F}$ be the set of fields E between L and K, i.e. $L \supseteq E \supseteq K$ and let $\mathcal{G}$ be the set of subgroups of G. Then, the maps $E \mapsto \mathrm{Gal}\,(L/E)$ and $H \mapsto L^H$ are inverses of each other and thus gives a one-to-one correspondence. Furthermore, for any field $E \in \mathcal{F}$, the extension E/K is Galois if and only if the corresponding Galois group $\mathrm{Gal}\,(L/E)$ is a normal subgroup of G. For the Galois extension E/K, $E \in \mathcal{F}$, the Galois group $\mathrm{Gal}\,(E/K)$ is (isomorphic to) the quotient group $G/\mathrm{Gal}\,(L/E)$.*

A *number field* is a finite extension of $\mathbb{Q}$. By the *degree of a number field* K we mean $[K : \mathbb{Q}]$, i.e. the degree of K as an extension of $\mathbb{Q}$. An *algebraic number* is a root of a polynomial over $\mathbb{Q}$ and an *algebraic integer* is an algebraic number whose minimal polynomial has integral coefficients. Let K be a number field. A *primitive element* η of K is an algebraic number such that $K = \mathbb{Q}(\eta)$. A polynomial $\mu(X)$ over $\mathbb{Q}$ is a *primitive polynomial* for K if $K = \mathbb{Q}[X]/\mu(X)$. By the primitive element theorem [Lang 1999, Theorem 4.6, Chapter V] every number field has a primitive element. Furthermore, we can assume that this primitive element is an algebraic integer [Lang 1999, Proposition 1.1, Chapter VII] (also refer to [van der Waerden 1991]). Thus any number field is isomorphic to the quotient $\mathbb{Q}[X]/\mu(X)$ where $\mu(X)$ is a monic irreducible polynomial with integral coefficients.

2.1. Input and Output Representations

The inputs and outputs of the algorithms we describe in this paper are objects like algebraic numbers, number fields, Galois groups etc. In this section we discuss suitable encodings of these objects into strings. The complexity of our algorithms are measured in terms of the sizes of these encodings. We use notation size (.) to denote the number of bits required to represent an object.

Integers are encoded in binary and hence for an integer n, its size is given by $\mathrm{size}\,(n) = \log n$. A rational number r is encoded as a pair of relatively prime integers a and b such that $r = \frac{a}{b}$. The size of r is therefore $O(\mathrm{size}\,(a){+}\mathrm{size}\,(b))$. A polynomial $T(X) = a_0{+}\ldots{+}a_n X^n \in \mathbb{Q}[X]$ is given by a list of its coefficients. Thus, $\mathrm{size}\,(T)$ is defined as $O\left(n + \sum_i \mathrm{size}\,(a_i)\right)$.

We now discuss how number fields are encoded. Recall that any number field can be expressed as a quotient $\mathbb{Q}[X]/\mu(X)$ where $\mu(X)$ is a primitive polynomial. We assume that a number field K is represented by giving a primitive polynomial $\mu(X)$ for it. In addition we will assume that $\mu(X)$ is monic with integral coefficients. Thus the size of K under this representation is the size of the polynomial $\mu(X)$. Notice that the size of K depends on the primitive polynomial chosen to represent K.

Let $K = \mathbb{Q}(\eta)$ be a number field of degree n represented via a primitive polynomial $\mu_\eta(X)$. Any algebraic number α in K can be expressed uniquely as $\alpha = A_\alpha(\eta)$ where $A_\alpha(X)$ is a polynomial over $\mathbb{Q}$ of degree less than n. By $\mathrm{size}\,(\alpha)$ we mean $\mathrm{size}\,(A_\alpha(X))$. Again the size of α depends on the chosen primitive element η of K. For a polynomial $f(X) = a_0 + \ldots + a_m X^m$ in $K[X]$ we define $\mathrm{size}\,(f)$ to be $\sum \mathrm{size}\,(a_i)$.

Let $f(X)$ be a polynomial of degree n over $\mathbb{Q}$. Landau's algorithm [Landau 1984] for computing its Galois group $\mathrm{Gal}\,(f)$, computes the entire multiplication table. Such an algorithm, in the worst case, will take time exponential in $\mathrm{size}\,(f)$ as the order of $\mathrm{Gal}\,(f)$ can be as large as $n!$. As mentioned in the introduction, we can consider $\mathrm{Gal}\,(f)$ as a permutation group on the roots of f and succinctly represent it by $n - 1$ many generating permutations [Luks 1993]. This gives $\mathrm{Gal}\,(f)$ a representation of size polynomial in n and it makes sense to ask if $\mathrm{Gal}\,(f)$ in this representation is computable in polynomial time. This is an open problem.

2.2. An algorithm for computing primitive elements

The proof of the primitive element theorem is algorithmic. Many algorithms in computational number theory make use of this algorithmic version. In order to keep the present paper self-contained, we give an algorithmic proof for a version of the theorem suitable for our purpose. We first recall a key lemma whose proof can be found in van der Waerden's book [van der Waerden 1991].

LEMMA 2.2. *Let α and β be algebraic numbers with conjugates $\alpha^{(i)}$, $1 \leq i \leq m$, and $\beta^{(j)}$, $1 \leq j \leq n$ respectively. Let $c \in \mathbb{Z}$ such that $\alpha^{(i)} + c\beta^{(j)} \neq \alpha^{(r)} + c\beta^{(r)}$ for all $(i,j) \neq (r,s)$ then $\mathbb{Q}(\alpha + c\beta) = \mathbb{Q}(\alpha, \beta)$. In particular, there is a positive integer $c \in \{1, 2, \ldots, m^2n^2 + 1\}$ such that $\mathbb{Q}(\alpha + c\beta) = \mathbb{Q}(\alpha, \beta)$.*

Next, we give an algorithm to compute minimal polynomial [Shoup 1999].

LEMMA 2.3. *Let α be an algebraic number with minimal polynomial $f(X) \in \mathbb{Q}[X]$. Given a polynomial $g(X) \in \mathbb{Q}[X]$ we can find the minimal polynomial for the element $g(\alpha) \in \mathbb{Q}(\alpha)$ in time polynomial in* $\text{size}\,(f) + \text{size}\,(g)$.

PROOF. Let the degree of f be n and let $\beta = g(\alpha)$. Our task is to compute the minimal polynomial $\mu_\beta(X)$ of β. Let m be the degree of the polynomial $\mu_\beta(X)$. Since β is an element of $\mathbb{Q}(\alpha)$, the degree m of its minimal polynomial is less than n. Furthermore, m is the least integer i less than n such that the set of powers $\{1, \beta, \ldots, \beta^{i-1}, \beta^i\}$ is *linearly dependent* as vectors over $\mathbb{Q}$. As $\beta = g(\alpha)$, the set $\{1, \ldots, \beta^i\}$ is linearly dependent if and only if there are $a_j \in \mathbb{Q}$ such that

$$g^i(X) + \sum_{j=0}^{i-1} a_j g^j(X) = 0 \mod f(X). \tag{1}$$

Equating the coefficients of X in equation 1 gives us a system of linear equations in a_j's, which can be checked for feasibility in time polynomial in $\text{size}\,(f) + \text{size}\,(g)$ using Gaussian elimination for example. Starting with $i = 0$ we find the least i for which equation 1 is feasible. Having found the least i, which is also the degree m of $\mu_\beta(X)$, we solve for the unknowns a_j. Clearly $\mu_\beta(X)$ is the polynomial $X^m + a_{m-1}X^{m-1} + \ldots + a_1X + a_0$. □

In the next lemma we prove an algorithmic version of the primitive element theorem.

LEMMA 2.4. *Let α be an algebraic number with minimal polynomial $f(X) \in \mathbb{Q}[X]$ of degree n. Let $\gamma_1, \ldots, \gamma_k$ be algebraic numbers in $\mathbb{Q}(\alpha)$ given as polynomials $\gamma_i = g_i(\alpha), 1 \leq i \leq k$, and let $K = \mathbb{Q}(\gamma_1, \ldots, \gamma_k)$ be the subfield of $\mathbb{Q}(\alpha)$ generated by $\gamma_1, \ldots, \gamma_k$. There is a deterministic algorithm with running time bounded by a polynomial in* $\text{size}\,(f) + \sum_{i=1}^{k} \text{size}\,(g_i)$ *that computes a polynomial $g(X) \in \mathbb{Q}[X]$ such that $g(\alpha)$ is a primitive element for K.*

PROOF. Consider the tower of fields $\mathbb{Q} \subseteq \mathbb{Q}(\gamma_1) \subseteq \ldots \subseteq \mathbb{Q}(\gamma_1, \ldots, \gamma_k) = K$. Our task is to compute a primitive element for K. For every $1 \leq i \leq k$, we will compute polynomials $h_i(X) \in \mathbb{Q}[X]$ such that $\eta_i = h_i(\alpha)$ is a primitive element of the subfield $\mathbb{Q}(\gamma_1, \ldots, \gamma_i)$.

For $i = 1$ choose $h_1(X) = g_1(X)$ and $\eta_1 = \gamma_1$. Inductively assume that we have computed $h_i(X)$ such that $\eta_i = h_i(\alpha)$ is a primitive element of the field $\mathbb{Q}(\gamma_1, \ldots, \gamma_i)$. Consider the field $\mathbb{Q}(\eta_i, \gamma_{i+1})$. As η_i and γ_{i+1} are elements of $\mathbb{Q}(\alpha)$, their degrees are less than n. By Lemma 2.2 there is an integer $c_{i+1} \in \{1, \ldots, n^4 + 1\}$ such that $\eta_i + c_{i+1}\gamma_{i+1}$ is a primitive element for the field $\mathbb{Q}(\eta_i, \gamma_{i+1}) = \mathbb{Q}(\gamma_1, \ldots, \gamma_{i+1})$.

We now explain how such a constant c_{i+1} can be computed. For a given c, to check whether $\eta_i + c\gamma_{i+1}$ is a primitive element of $\mathbb{Q}(\eta_i, \gamma_{i+1})$ it suffices to check whether there are polynomials $A(X)$ and $B(X)$ of degree at most n such that $A(\eta_i + c\gamma_{i+1}) = \eta_i$ and $B(\eta_i + c\gamma_{i+1}) = \gamma_{i+1}$. This can be done by checking whether the following equations are feasible for unknowns a_j and b_j.

$$\sum_{j=0}^{n-1} a_j(h_i(X) + cg_{i+1}(X))^j \;=\; h_i(X) \mod f(X)$$

$$\sum_{j=0}^{n-1} b_j(h_i(X) + cg_{i+1}(X))^j \;=\; g_{i+1}(X) \mod f(X)$$

Equating the coefficients of X, this involves checking feasibility of a system of linear equations over $\mathbb{Q}$ and thus can be done in time polynomial in $\text{size}\,(h_i)+\text{size}\,(g_{i+1})+\text{size}\,(f)$. We go over all $1 \leq c \leq n^4 + 1$ and let c_{i+1} be the least c for which the above equations are feasible. The required polynomial $h_{i+1}(X)$ is given by $h_{i+1}(X) = h_i(X) + c_{i+1}g_{i+1}(X)$. Finally, $h_k(\alpha)$ is a primitive element for K. The overall running time for the algorithm is bounded by a polynomial in the size of $\text{size}\,(f) + \sum_{j=1}^{k} \text{size}\,(g_j)$. □

Remark 2.5. A similar algorithm can be designed that takes as input the minimal polynomials $g_i(X) \in \mathbb{Q}[X]$ of the algebraic numbers γ_i respectively, for $1 \leq i \leq k$ and computes a primitive element for the field $K = \mathbb{Q}(\gamma_1, \ldots, \gamma_k)$. The running time for this algorithm is polynomial in $[K : \mathbb{Q}]$ and the sizes of the minimal polynomials $g_i(X)$.

3. PREVIOUS COMPLEXITY RESULTS

We now state some of the known results in computational Galois theory formally. The following result for computing the Galois group of a polynomial $f(X)$ that runs in time polynomial in the size of the Galois group and $f(X)$ is due to Landau [Landau 1984].

THEOREM 3.1 (LANDAU). *There is a deterministic algorithm that takes as input a number field K, a polynomial $f(X) \in K[X]$ and a positive integer b in unary, and in time bounded by* $\text{size}\,(f)$, $\text{size}\,(K)$ *and b, decides if* $\text{Gal}\,(K_f/K)$ *has at most b elements, and if so computes* $\text{Gal}\,(K_f/K)$ *by finding the entire multiplication table of* $\text{Gal}\,(K_f/K)$ *(and hence also by giving the generating set of* $\text{Gal}\,(K_f/K)$ *as a permutation group on the roots of $f(X)$).*

The algorithm first computes a primitive element θ of K_f. Determining $\text{Gal}\,(f)$ amounts to finding the action of the automorphisms on θ. Subsequently, Landau and Miller [Landau and Miller 1985] gave their polynomial-time solvability test.

THEOREM 3.2 (LANDAU-MILLER). *There is a deterministic polynomial-time algorithm that takes as input a polynomial $f(X) \in \mathbb{Q}[X]$ and tests if the Galois group* $\text{Gal}\,(f)$ *of f is solvable.*

A byproduct of the Landau-Miller algorithm is an algorithm to compute the primes that divide the order of the Galois group. We summaries this result for use latter on.

THEOREM 3.3 (LANDAU-MILLER). *There is a deterministic polynomial-time algorithm that takes as input a polynomial $f(X)$ over $\mathbb{Q}$ and, if $f(X)$ is solvable by radicals, computes the prime factors of* $\#\text{Gal}\,(f)$.

Thus one can also check in deterministic polynomial time whether the Galois group is a p-group.

4. GROUP THEORETICAL PRELIMINARIES

We recall some group theory. Details can be found in Marshall Hall's text [Hall Jr. 1959, Chapter 10]. Let G be any group. The *lower central series* of G is the sequence of groups $G = G_0 \geq G_1 \ldots \geq G_n \geq \ldots$ where $G_{i+1} = [G_i, G]$. A group G is said to be *nilpotent* if its lower central series is of finite length, i.e. $G_c = \{1\}$ for some nonnegative integer c, where 1 denotes the identity element. The least c such that $G_c = \{1\}$ is called the *class of nilpotence* of a nilpotent group G. If the group G is finite then it is nilpotent if and only if all its p-Sylow subgroups are normal. It follows from Sylow's theorem that a finite nilpotent group is a product of its p-Sylow subgroups. It is this characterisation of finite nilpotent groups that will be useful for us in our nilpotence test.

We recall some permutation group theory from Wielandt's book [Wielandt 1964]. Let Ω be a finite set. The *symmetric group* $\mathrm{Sym}(\Omega)$ is the group of all permutations on Ω. By a *permutation group on* Ω we mean a subgroup of $\mathrm{Sym}(\Omega)$. For $\alpha \in \Omega$ and $g \in \mathrm{Sym}(\Omega)$, let α^g denote the image of α under the permutation g. For $A \subseteq \mathrm{Sym}(\Omega)$, α^A denotes the set $\{\alpha^g : g \in A\}$. In particular, for $G \leq \mathrm{Sym}(\Omega)$ the *G-orbit* containing α is α^G. The G-orbits form a partition of Ω. Given $G \leq \mathrm{Sym}(\Omega)$ by a generating set A and $\alpha \in \Omega$, there is a polynomial-time algorithm to compute α^G [Luks 1993].

Sometimes we need to consider a more general group action on a set Ω. In the generalised setting, we say G *acts* on Ω if there is a group homomorphism $\varphi : G \longrightarrow \mathrm{Sym}(\Omega)$. The kernel $Ker(\varphi)$ of this action is the subgroup of G whose image under φ is the identity element (which pointwise fixes Ω). If $Ker(\varphi)$ is trivial we say that the action is *faithful.* In this paper, when we say G is a permutation group on a set Ω we mean a faithful action unless explicitly stated. The only exceptions arise when we restrict the group G to a subset of Ω, typically an orbit or a block.

For $\Delta \subseteq \Omega$ and $g \in \mathrm{Sym}(\Omega)$, Δ^g denotes $\{\alpha^g : \alpha \in \Delta\}$. The set-wise stabilizer of Δ, i.e. $\{g \in G : \Delta^g = \Delta\}$, is denoted by G_Δ. If Δ is the singleton set $\{\alpha\}$ we write G_α instead of $G_{\{\alpha\}}$. An often used result is the orbit-stabilizer formula stated below [Wielandt 1964, Theorem 3.2].

THEOREM 4.1 (ORBIT-STABILIZER FORMULA). *Let $G \leq \mathrm{Sym}(\Omega)$ be a permutation group and let α be any element of Ω then the order of the group G is given by $\#G = \#G_\alpha \cdot \#\alpha^G$.*

A permutation group $G \leq \mathrm{Sym}(\Omega)$ is *transitive* if there is a single G-orbit. Suppose $G \leq \mathrm{Sym}(\Omega)$ is transitive. Then a non-empty subset Δ of Ω is a *G-block* if for all $g \in G$ either $\Delta^g = \Delta$ or $\Delta^g \cap \Delta = \emptyset$. For every G, Ω is a block and each singleton $\{\alpha\}$ is a block. These are the *trivial blocks* of G. A transitive group G is *primitive* if it has only trivial blocks and it is *imprimitive* if it has nontrivial blocks. We state a useful proposition that is easy to prove from these definitions.

PROPOSITION 4.2. *Let $G \leq \mathrm{Sym}(\Omega)$ be a transitive permutation group.*

(a) If $\Delta \subset \Omega$ is a G-block then G_Δ is transitive on Δ. I.e. for $\alpha, \beta \in \Delta$ there is a $g \in G_\Delta$ such that $\alpha^g = \beta$.
(b) Let $\Sigma \subset \Omega$ be a G-block. Then $\Delta \subset \Sigma$ is a G-block if and only if Δ is a G_Σ-block.

A G-block Δ is a *maximal subblock* of a G-block Σ if $\Delta \subset \Sigma$ and there is no G-block Υ such that $\Delta \subset \Upsilon \subset \Omega$. Let Δ and Σ be two G-blocks. A chain $\Delta = \Delta_0 \subset \ldots \subset \Delta_t = \Sigma$ is a *maximal chain* of G-blocks between Δ and Σ if for all i, Δ_i is a maximal subblock of Δ_{i+1}.

For a G-block Δ and $g \in G$, Δ^g is also a G-block and $\#\Delta = \#\Delta^g$. Let Δ and Σ be two G-blocks such that $\Delta \subseteq \Sigma$. The *Δ-block system of* Σ, is the collection

$$\mathcal{B}(\Sigma/\Delta) = \{\Delta^g : g \in G \text{ and } \Delta^g \subseteq \Sigma\}.$$

The set $\mathcal{B}(\Sigma/\Delta)$ is a partition of Σ. It follows that $\#\Delta$ divides $\#\Sigma$ and by *index* of Δ in Σ, which we denote by $[\Sigma : \Delta]$, we mean $\#\mathcal{B}(\Sigma/\Delta) = \frac{\#\Sigma}{\#\Delta}$. We will use $\mathcal{B}(\Delta)$ to denote $\mathcal{B}(\Omega/\Delta)$. We state the connection between blocks and subgroups [Wielandt 1964, Theorem 7.5].

THEOREM 4.3 (GALOIS CORRESPONDENCE OF BLOCKS). *Let $G \leq \mathrm{Sym}(\Omega)$ be transitive and $\alpha \in \Omega$. For $G \geq H \geq G_\alpha$ the orbit $\Delta = \alpha^H$ is a G-block and $G_\Delta = H$. The correspondence $\alpha^H = \Delta \rightleftharpoons G_\Delta = H$ is a one-to-one correspondence between G-blocks Δ containing α and subgroups H of G containing G_α. Furthermore for G-blocks $\Delta \subseteq \Sigma$ we have $[G_\Sigma : G_\Delta] = [\Sigma : \Delta]$.*

Let $G \leq \mathrm{Sym}(\Omega)$ be transitive and Δ and Σ be two G-blocks such that $\Delta \subseteq \Sigma$. Let $\mathrm{G}(\Sigma/\Delta)$ denote the group $\{g \in G : \Upsilon^g = \Upsilon \text{ for all } \Upsilon \in \mathcal{B}(\Sigma/\Delta)\}$. We write G^Δ for the group $\mathrm{G}(\Omega/\Delta)$. The next three lemmas are well known in the permutation group theory community, however we prove them here for completeness and to fix the notation.

LEMMA 4.4. *Let $G \leq \mathrm{Sym}(\Omega)$ be a permutation group and let Δ and Σ be two G-blocks such that $\Sigma \supseteq \Delta$. Then*

(*1*) *The group $\mathrm{G}(\Sigma/\Delta)$ is the largest normal subgroup of G_Σ contained in G_Δ. In particular, G^Δ is a normal subgroup of G.*
(*2*) *The quotient group $G_\Sigma/\mathrm{G}(\Sigma/\Delta)$ is a faithful permutation group on $\mathcal{B}(\Sigma/\Delta)$ and is primitive when Δ is a maximal subblock.*

PROOF. For any $g \in G_\Sigma$, since g set-wise stabilises Σ, g permutes the elements of $\mathcal{B}(\Sigma/\Delta)$. Hence for any $\Upsilon \in \mathcal{B}(\Sigma/\Delta)$ we have $\Upsilon^{g^{-1}\mathrm{G}(\Sigma/\Delta)g} = \Upsilon$. Thus, $\mathrm{G}(\Sigma/\Delta)$ is a normal subgroup of G_Σ.

Now consider any $N \subseteq G_\Delta$ which is a normal subgroup of G_Σ. Since $\Delta^N = \Delta$, and since G_Σ acts transitively on $\mathcal{B}(\Sigma/\Delta)$, for any $\Upsilon \in \mathcal{B}(\Sigma/\Delta)$ there is a $g \in G_\Sigma$ such that $\Upsilon = \Delta^g$. Therefore, $\Upsilon^N = \Delta^{gN} = \Delta^{Ng} = \Upsilon$ for each $\Upsilon \in \mathcal{B}(\Sigma/\Delta)$. Thus $N \subseteq \mathrm{G}(\Sigma/\Delta)$. Since $\mathrm{G}(\Sigma/\Delta) \trianglelefteq G_\Sigma$ we have proved part 1.

Consider the action of G_Σ on $\mathcal{B}(\Sigma/\Delta)$. Clearly, $\mathrm{G}(\Sigma/\Delta)$ is the kernel in this action. Therefore $G_\Sigma/\mathrm{G}(\Sigma/\Delta)$ acts faithfully on $\mathcal{B}(\Sigma/\Delta)$. Notice that G_Σ is transitive on $\mathcal{B}(\Sigma/\Delta)$ as it is transitive on Σ. Further it can be easily verified that nontrivial $G_\Sigma/\mathrm{G}(\Sigma/\Delta)$-blocks of $\mathcal{B}(\Sigma/\Delta)$ are in 1-1 correspondence with the G-blocks Γ such that $\Delta \subset \Gamma \subset \Sigma$. Thus, $G_\Sigma/\mathrm{G}(\Sigma/\Delta)$ is primitive if and only if Δ is a maximal subblock of Σ.

□

In our algorithms, we often need to check certain properties of the groups G_Δ and G^Δ. The group G in this context is the Galois group of the input polynomial f and Δ is a G-block on its action on the roots of f. However, explicit computation of these groups are impossible in polynomial time. The next lemma helps us reduce this problem to the study of certain natural quotient groups.

LEMMA 4.5. *Let $G \leq \mathrm{Sym}(\Omega)$ be a permutation group. Let Δ and Σ be two G-block such that $\Delta \subseteq \Sigma$. Then the quotient group G^Σ/G^Δ can be embedded* into *the product group $\left(G_\Sigma/\mathrm{G}(\Sigma/\Delta)\right)^l$ for some positive integer l.*

PROOF.
For the proof, let the Σ-block system $\mathcal{B}(\Sigma)$ be $\{\Sigma_1, \ldots, \Sigma_m\}$ where $\Sigma = \Sigma_1$. Notice that $G^{\Sigma_i} = G^\Sigma = G^{\Sigma_j}$ for all $1 \leq i \leq j \leq m$. Let $\Delta = \Delta_1, \ldots, \Delta_m$ be any m elements of the Δ-block system $\mathcal{B}(\Delta)$ such that $\Delta_i \subseteq \Sigma_i$.

Consider the action of G^Σ on $\mathcal{B}(\Delta)$. Clearly the kernel of this action is G^Δ. Therefore, the quotient group $H = G^\Sigma/G^\Delta$ acts faithfully on $\mathcal{B}(\Delta)$. Notice that the orbits under the action are precisely $\mathcal{B}(\Sigma_i/\Delta_i)$ for $1 \leq i \leq m$. It is thus easy to see that H can be embedded in the product $\prod H_i$ where H_i is the restriction of H on to $\mathcal{B}(\Sigma_i/\Delta_i)$. Notice that $G^\Sigma \subseteq G_{\Sigma_i}$ for all i and the kernel of the action of G_{Σ_i} on $\mathcal{B}(\Sigma_i/\Delta_i)$ is $\mathrm{G}(\Sigma_i/\Delta_i)$ (Lemma 4.4). Therefore, G^Σ/G^Δ can be embedded *into* a subgroup of the product group $\prod_i G_{\Sigma_i}/\mathrm{G}(\Sigma_i/\Delta_i)$. The lemma then follows from the fact that G_{Σ_i} is isomorphic to G_Σ (in fact G_{Σ_i} and G_Σ are G-conjugates) and $\mathrm{G}\left(\Sigma/\Delta\right)$ is isomorphic to $\mathrm{G}\left(\Sigma_i/\Delta_i\right)$.

□

The next lemma connects orbits of normal subgroups and blocks.

LEMMA 4.6. *Let $G \leq \mathrm{Sym}\left(\Omega\right)$ be transitive and $N \trianglelefteq G$. Let $\alpha \in \Omega$. Then the N-orbit α^N is a G-block and the collection of N-orbits is an α^N-block system of Ω under G action. If $N \neq \{1\}$ then $\#\alpha^N > 1$. Furthermore, if $G_\alpha \leq N \neq G$ then the α^N-block system is nontrivial implying that G is not primitive.*

PROOF. Let $\alpha \in \Omega$ and $g \in G$. If $\beta = \alpha^g$ then the set $(\alpha^N)^g = \alpha^{Ng} = \alpha^{gN} = \beta^N$. Thus $(\alpha^N)^g$ and α^N are N-orbits, and hence are identical or disjoint. So, α^N is a G-block and the N-orbits form the α^N-block system of G. If $\#\alpha^N = 1$ then all the N orbits are of cardinality 1 and hence N fixes all element of Ω. This is possible if and only if $N = 1$.

Finally, suppose that $G_\alpha \leq N$. Then by the Orbit-Stabilizer formula (Theorem 4.1) $\#G = \#\Omega \cdot \#G_\alpha$ and $\#N = \#\alpha^N \cdot \#G_\alpha$. Thus, if $\{1\} \neq N \neq G$ and $G_\alpha \leq N$ then $1 < \#\alpha^N < \#\Omega$ and hence α^N is a nontrivial G-block. □

5. NILPOTENT PERMUTATION GROUPS

In this section, we prove some properties of transitive nilpotent groups that will be required for our nilpotence test. Recall that a finite group G is nilpotent if and only if for all prime factors p of $\#G$, there is a unique normal p-Sylow subgroup for G. Let G_p denote this unique p-Sylow subgroup of G. Any orbit of the Sylow subgroup G_p is a G-block as G_p is normal in G (Lemma 4.6). These blocks play a crucial role in our nilpotence test and hence we give them a name.

Definition 5.1 (*Sylow blocks*). Let G be a transitive nilpotent permutation group on Ω and let p be a prime dividing the order of G. A subset of Ω is called a *p-Sylow block* of G if it is an orbit of the p-Sylow subgroup G_p.

We prove the following lemma about Sylow blocks.

LEMMA 5.2. *Let $G \leq \mathrm{Sym}\left(\Omega\right)$ be a transitive nilpotent permutation group. For every prime p that divides $\#G$ any p-Sylow block is of cardinality p^l for some $l > 0$.*

PROOF. Let $\Sigma \subseteq \Omega$ be any p-Sylow block. Since Σ is an orbit of a nontrivial normal subgroup G_p of G, by Lemma 4.6, we have $\#\Sigma > 1$. Furthermore if α is any element in Σ, since $\Sigma = \alpha^{G_p}$ by the orbit-stabiliser formula (Theorem 4.1) we have $\#\Sigma \cdot \#(G_p)_\alpha = \#G_p$. Hence $\#\Sigma$ is a power of p. So, $\#\Sigma = p^l$ for some $l > 0$. □

We now prove the following lemma about the cardinality of Sylow blocks.

LEMMA 5.3. *Let $G \leq \mathrm{Sym}\left(\Omega\right)$ be a transitive nilpotent permutation group. For any prime p, p divides $\#G$ if and only if it divides $\#\Omega$, and for any p-Sylow block Σ of G, $\#\Sigma$ is the highest power of p that divides $\#\Omega$.*

PROOF. Consider any $\alpha \in \Omega$. Since G is transitive, we known that $\alpha^G = \Omega$. By the orbit-stabiliser theorem (Theorem 4.1) we have $\#\Omega \cdot \#G_\alpha = \#G$. Therefore, every prime factor of $\#\Omega$ divides $\#G$. Conversely, if p divides $\#G$ then for any p-Sylow block Σ, p divides $\#\Sigma$ by Lemma 5.2. Hence p divides $\#\Omega$ (a consequence of Theorem 4.3 for blocks Ω and Σ).

We now prove that the cardinality of the p-Sylow block Σ is the highest power of p that divides $\#\Omega$. Consider the blocks Ω and Σ. By Theorem 4.3 we have $[\Omega : \Sigma] = [G : G_\Sigma]$.

Since Σ is a G_p orbit we have $G_p \leq G_\Sigma$ and hence p does not divide $[G : G_\Sigma]$. So, p does not divide $[\Omega : \Sigma] = \frac{\#\Omega}{\#\Sigma}$. □

As mentioned before, the Sylow blocks play an important role in our algorithm. The fact that any Sylow block Σ has prime power cardinality helps us in studying the Sylow subgroups of G_Σ. We prove the following lemma about any block of prime power cardinality.

LEMMA 5.4. *Let $G \leq \mathrm{Sym}(\Omega)$ be a transitive nilpotent permutation group. Let Δ be any G-block such that $\#\Delta$ is a power of a prime p and let $q \neq p$ be another prime dividing $\#G$. Let $G_{\Delta,q}$ denote the q-Sylow subgroup of G_Δ. Then $G_{\Delta,q}$ fixes all points of the block Δ (i.e. for all g in $G_{\Delta,q}$ and α in Δ, $\alpha^g = \alpha$). As a result the q-Sylow subgroups $G_{\Delta,q}$ and $G_{\alpha,q}$ are equal.*

PROOF. Let $\alpha \in \Delta$. Consider the blocks Δ and $\{\alpha\}$. By Theorem 4.3 we have $G_\alpha \leq G_\Delta$, and the index $[G_\Delta : G_\alpha] = \#\Delta$ is a power of p. Consequently, for a prime $q \neq p$ the highest power of q that divides $\#G_\alpha$ and $\#G_\Delta$ are same, say q^r. Further, note that both G_Δ and G_α are nilpotent as they are subgroups of a nilpotent group G. Therefore, they have unique q-Sylow subgroups $G_{\Delta,q}$ and $G_{\alpha,q}$, respectively which must be of size q^r. Hence $G_{\Delta,q} = G_{\alpha,q}$, implying that $\alpha^g = \alpha$ for all $g \in G_{\alpha,q}$. □

We derive an important consequence of Lemma 5.4.

LEMMA 5.5. *Let $G \leq \mathrm{Sym}(\Omega)$ be a transitive nilpotent permutation group and Σ be any p-Sylow block of G. The group G^Σ is the (unique) p-Sylow subgroup G_p of G.*

PROOF. Recall that Σ is an orbit of G_p and the Σ-block system $\mathcal{B}(\Sigma)$ is the collection of all G_p-orbits and hence all p-Sylow blocks of G. Therefore, for all $\Gamma \in \mathcal{B}(\Sigma)$, we have $\Gamma^{G_p} = \Gamma$ and hence $G_p \leq G^\Sigma$. The group G^Σ being a subgroup of a nilpotent group G is itself nilpotent and is therefore the product of its Sylow subgroups. So, to prove that $G^\Sigma = G_p$ it is sufficient to prove that for all primes $q \neq p$ the q-Sylow subgroup G^Σ_q of G^Σ is trivial.

Note that $G^\Sigma = \bigcap_{\Gamma \in \mathcal{B}(\Sigma)} G_\Gamma$. Hence, $G^\Sigma_q \leq G_{\Gamma,q}$ for all $\Gamma \in \mathcal{B}(\Sigma)$ and by Lemma 5.4 it follows that for all g in G^Σ_q and $\alpha \in \Gamma$ we have $\alpha^g = \alpha$. Since $\bigcup_{\Gamma \in \mathcal{B}(\Sigma)} \Gamma = \Omega$, for all α in Ω and g in G^Σ_q, $\alpha^g = \alpha$. This is only possible if G^Σ_q is the trivial group $\{1\}$. □

We now show that the subblock structure of G under a p-Sylow block is similar to the subblock structure of a transitive p-group.

LEMMA 5.6. *Let $G \leq \mathrm{Sym}(\Omega)$ be a transitive nilpotent permutation group and let p be a prime factor of $\#G$. Let Σ be any p-Sylow block of G then for any subset $\Delta \subseteq \Sigma$, Δ is a G-block if and only if Δ is G_p-block under the transitive action of G_p on Σ.*

PROOF. Clearly if $\Delta \subseteq \Sigma$ is a G-block then it is also a G_p-block as $G_p \leq G$.

Conversely, suppose Δ is a G_p-block. We first argue that Δ is a G_Σ-block. Recall that G_p is the p-Sylow subgroup of G_Σ as well, and by Lemma 5.4 for a prime $q \neq p$ the q-Sylow subgroups of G_Σ pointwise fix each element of Σ (and hence each element of Δ). Since G_Σ is a product of its Sylow subgroups, it follows that Δ is a G_Σ-block. Hence, by Proposition 4.2 Δ is a G-block as well. □

The previous lemma indicates that to study the subblock structure under a p-Sylow block it is sufficient to understand the subblock structure of a transitive p-group. We now recall a result about blocks of transitive p-groups (see e.g. Luks [Luks 1982, Lemma 1.1]).

LEMMA 5.7. *Let $G \leq \mathrm{Sym}(\Omega)$ be a transitive permutation p-group and Δ be a maximal G-block. Then $[\Omega : \Delta] = p$ and the group G_Δ is a normal subgroup of G with the quotient group G/G_Δ being the cyclic group of order p.*

The above lemma has the following corollary.

Corollary 5.8. *Let G be a transitive permutation p-group on Ω and let Δ be any G-block. Suppose that Γ is a minimal G-block containing Δ then $[\Gamma : \Delta] = p$ and G_Δ is a normal subgroup of G_Γ with quotient G_Γ/G_Δ a cyclic group of order p.*

Proof. Consider the transitive action of G_Γ on Γ. If Δ' is any G_Γ-block between Γ and Δ then $G_\Delta \leq G_{\Delta'} \leq G_\Gamma$. Hence by Theorem 4.3, Δ' is a G-block. This contradicts the minimality of Γ. So, Δ is a maximal G_Γ block of Γ. The corollary then follows from Lemma 5.7 for the group G_Γ. □

We now translate the above result on the block structure of a p-group into that of a nilpotent group.

Theorem 5.9. *Let G be any transitive nilpotent group on Ω and let p be a prime dividing $\#G$ (and hence $\#\Omega$). Let p^m be the highest power of p that divides $\#\Omega$ and let Δ be a G-block of cardinality a power of p. Then there is a p-Sylow block Σ such that $\Delta \subseteq \Sigma$. Furthermore, if $\#\Delta < p^m$ then for any minimal G-block Δ' such that $\Delta \subset \Delta' \subseteq \Sigma$ we have:*

(*1*) *The index of the blocks $[\Delta' : \Delta]$ is p,*
(*2*) *The group G_Δ is a normal subgroup of $G_{\Delta'}$ and*
(*3*) *The quotient group $G_{\Delta'}/G_\Delta$ is the cyclic group of order p.*

Proof. First we show that Δ is a subset of a p-Sylow block. Let α be any element of Δ and let Σ be the p-Sylow block α^{G_p}. We claim that $\Delta \subseteq \Sigma$. By Proposition 4.2, $\Delta = \alpha^{G_\Delta}$. By Lemma 5.4, for any prime $q \neq p$ the q-Sylow subgroup $G_{\Delta,q}$ of G_Δ fixes each point of Δ. Since G_Δ is the product of its Sylow subgroups, it follows that $\Delta = \alpha^{G_\Delta} = \alpha^{G_{\Delta,p}} \subseteq \alpha^{G_p} = \Sigma$.

Now consider any minimal G-block Δ' between Σ and Δ. By Lemma 5.6, Δ' is a minimal G_p-block between Δ and Σ. Therefore, using Corollary 5.8, we have $[\Delta' : \Delta] = p$ and $G_{\Delta,p}$ is a normal subgroup of $G_{\Delta',p}$ such that their quotient group $G_{\Delta',p}/G_{\Delta,p}$ is a cyclic group of order p. For all primes q different from p by Lemma 5.4 we have $G_{\Delta',q} = G_{\Delta,q}$. Since $G_{\Delta'}$ and G_Δ are a product of their Sylow subgroups we have G_Δ is a normal subgroup of $G_{\Delta'}$ with the quotient $G_{\Delta'}/G_\Delta = G_{\Delta',p}/G_{\Delta,p}$, a cyclic group of order p. □

Theorem 5.10. *Let G be a transitive permutation group on Ω. Let α be any element of Ω. Suppose that for all primes p dividing $\#G$ we have a chain $\{\alpha\} = \Delta_0 \subset \ldots \subset \Delta_m$ of G-blocks satisfying the follow properties*

(*1*) *The index $[\Delta_{i+1} : \Delta_i] = p$,*
(*2*) *The group G_{Δ_i} is a normal subgroup of $G_{\Delta_{i+1}}$ and*
(*3*) *The prime p does not divide the order of G/G^{Δ_m}.*

Then G is nilpotent.

Proof. For each prime factor p of $\#G$ we will show that any p-Sylow subgroup of G is normal. Since G^{Δ_m} is normal in G (by Lemma[Part 1] 4.4) and by part 3, p does not divide $\#G/G^{\Delta_m}$, it is sufficient to prove that G^{Δ_m} is a p-group. We prove inductively that for all $0 \leq i \leq m$ the group G^{Δ_i} is of cardinality p^{l_i} for some l_i. As the base case, $G^{\Delta_0} = \{1\}$ and hence $\#G^{\Delta_0} = p^{l_0}$ where $l_0 = 0$.

Suppose that our hypothesis is true for all $i \leq r$. To prove that $\#G^{\Delta_{r+1}}$ is $p^{l_{r+1}}$ for some l_{r+1} it is sufficient to prove that the quotient group $G^{\Delta_{r+1}}/G^{\Delta_r}$ is a p-group since by the inductive assumption $\#G^{\Delta_r} = p^{l_r}$. From Lemma 4.5 the quotient group $G^{\Delta_{r+1}}/G^{\Delta_r}$ is a subgroup of $\left(G_{\Delta_{r+1}}/\mathrm{G}\left(\Delta_{r+1}/\Delta_r\right)\right)^l$ for some integer l. We will prove that $G_{\Delta_{r+1}}/\mathrm{G}\left(\Delta_{r+1}/\Delta_r\right)$ is a cyclic group of order p which clearly is sufficient.

The group $\mathrm{G}\left(\Delta_{r+1}/\Delta_r\right)$ is the largest subgroup of G_{Δ_r} that is normal in $G_{\Delta_{r+1}}$ (Lemma 4.4). By part 2, G_{Δ_r} itself is normal in $G_{\Delta_{r+1}}$. Hence the groups $\mathrm{G}\left(\Delta_{r+1}/\Delta_r\right)$ and G_{Δ_r} are equal. Furthermore, $[G_{\Delta_{r+1}} : G_{\Delta_r}] = [\Delta_{r+1} : \Delta_r] = p$ (part 1). So, the quotient group $G_{\Delta_{r+1}}/\mathrm{G}\left(\Delta_{r+1}/\Delta_r\right)$, which is $G_{\Delta_{r+1}}/G_{\Delta_r}$, is a cyclic group of order p. □

Remark 5.11. Both Theorems 5.9 and 5.10 play an important role in our algorithm described in the next section. Theorem 5.9 guarantees for nilpotent groups that each chain of G-blocks (whose sizes are a power of p) can be extended to a maximal chain that terminates at a p-Sylow block and any pair of adjacent blocks have index p. This allows the algorithm to grow the chain of blocks in any manner. Theorem 5.10 ensures the correctness.

6. THE POLYNOMIAL-TIME NILPOTENCE TEST

In this section, our goal is to give an algorithm that takes as input a polynomial $f(X)$ over $\mathbb{Q}$ and checks whether its Galois group $\mathrm{Gal}\,(f)$ is nilpotent. Now, $\mathrm{Gal}\,(f)$ is nilpotent if and only if for each irreducible factor $h(X)$ of $f(X)$, the Galois group $\mathrm{Gal}\,(h)$ is nilpotent. This is true because nilpotent groups are closed under subgroups, products and quotients. Hence, in order to test the nilpotence of $Gal f$ it suffices to check for each irreducible factor h of f that its Galois group $\mathrm{Gal}\,(h)$ is nilpotent. Furthermore, using the LLL algorithm [Lenstra et al. 1982], all the irreducible factors of $f(X)$ can be computed in polynomial time. Therefore, for nilpotence testing, we assume without loss of generality that the input polynomial $f(X)$ is irreducible.

Let G be $\mathrm{Gal}\,(f)$. We consider G as a subgroup of $\mathrm{Sym}\,(\Omega)$, where Ω is the set of roots of $f(X)$. Since f is irreducible all its roots are distinct, and for any two roots α and β of f there is an element σ in the Galois group G of f such $\alpha^\sigma = \beta$. Therefore, the Galois group G is a transitive subgroup of $\mathrm{Sym}\,(\Omega)$.

We first outline the main idea. For all primes p dividing $\#G$ if we can test the existence of a tower of G-blocks $\{\alpha\} = \Delta_0 \subseteq \ldots \subset \Delta_m$ satisfying the conditions of Theorem 5.10 then G is nilpotent. It is not clear whether we can test these conditions by explicitly computing the G-blocks. That seems possible only if we can already compute the Galois group G. Instead, our approach will be to test the conditions of Theorem 5.10 by considering the fixed field of each group G_{Δ_i}. For a G-block Δ, let $\mathbb{Q}_\Delta$ denote the fixed field of the splitting field $\mathbb{Q}_f$ under the automorphisms in G_Δ. More precisely,

$$\mathbb{Q}_\Delta = \{\beta \in \mathbb{Q}_f \mid \beta^g = \beta \text{ for all } g \in G_\Delta\}.$$

The following proposition is a consequence of the fundamental theorem of Galois theory (Theorem 2.1).

PROPOSITION 6.1. *Let $\mathbb{Q}_\Delta$ denote the fixed field of $\mathbb{Q}_f$ under automorphisms in G_Δ. Then:*

(*1*) *The Galois group* $\mathrm{Gal}\,(\mathbb{Q}_f/\mathbb{Q}_\Delta)$ *is the group G_Δ.*
(*2*) *Let α be any root of the polynomial $f(X)$ such that $\alpha \in \Delta$ then $\mathbb{Q}_\Delta$ is a subfield of $\mathbb{Q}(\alpha)$.*
(*3*) *If $\mu_\Delta(X)$ is a primitive polynomial for $\mathbb{Q}_\Delta$ then its Galois group* $\mathrm{Gal}\,(\mu_\Delta)$ *is G/G^Δ.*

PROOF. Part 1 follows directly from the fundamental theorem of Galois theory as $\mathbb{Q}_\Delta$ of is the fixed field of $\mathbb{Q}_f$ under G_Δ. To prove that $\mathbb{Q}_\Delta \subseteq \mathbb{Q}(\alpha)$ notice that the Galois groups $\mathrm{Gal}\,(\mathbb{Q}_f/\mathbb{Q}_\Delta)$ and $\mathrm{Gal}\,(\mathbb{Q}_f/\mathbb{Q}(\alpha))$ are G_Δ and G_α respectively. As $\alpha \in \Delta$ the group G_α is a subgroup of G_Δ. Hence by the fundamental theorem of Galois theory $\mathbb{Q}_\Delta \subseteq \mathbb{Q}(\alpha)$.

For the third part, notice that if $\mu_\Delta(X)$ is the primitive polynomial of $\mathbb{Q}_\Delta$ then its splitting field $\mathbb{Q}_{\mu_\Delta}$ is the normal closure of $\mathbb{Q}_\Delta$ in $\mathbb{Q}_f$. Hence, the Galois group $\mathrm{Gal}\,(\mathbb{Q}_f/\mathbb{Q}_{\mu_\Delta})$ is the largest normal subgroup of G that is contained in G_Δ. By Lemma 4.4 (putting $\Sigma = \Omega$ in the lemma) it follows that the Galois group $\mathrm{Gal}\,(\mathbb{Q}_f/\mathbb{Q}_{\mu_\Delta})$ is precisely G^Δ. Consequently, the Galois group $\mathrm{Gal}\,(\mu_\Delta) = \mathrm{Gal}\,(\mathbb{Q}_{\mu_\Delta}/\mathbb{Q})$ is G/G^Δ. □

A direct consequence of Proposition 6.1 and Theorem 2.1 is the following.

PROPOSITION 6.2. *A tower of G-blocks $\{\alpha\} = \Delta_0 \subset \ldots \subset \Delta_m$ satisfies the conditions of Theorem 5.10 if and only if the tower of fixed field $\mathbb{Q}(\alpha) = \mathbb{Q}_{\Delta_0} \supset \ldots \supset \mathbb{Q}_{\Delta_m}$ satisfies the following conditions:*

(*1*) *The degree of the extension $\mathbb{Q}_{\Delta_i}/\mathbb{Q}_{\Delta_{i+1}}$ is p.*
(*2*) *The extension $\mathbb{Q}_{\Delta_i}/\mathbb{Q}_{\Delta_{i+1}}$ is normal.*
(*3*) *For any block Δ if μ_Δ denote a primitive polynomial of the field $\mathbb{Q}_\Delta$, then the prime p does not divide the order of the Galois group $\mathrm{Gal}\,(\mu_{\Delta_m})$ of the primitive polynomial $\mu_{\Delta_m}(X)$ of $\mathbb{Q}_{\Delta_m}$.*

We will first check whether $f(X)$ is solvable by radicals by using the Landau-Miller test. Clearly, if $f(X)$ is not solvable by radicals then G is not nilpotent. If $f(X)$ is solvable by radicals then so is each polynomial $\mu_{\Delta_i}(X)$ for $1 \le i \le m$. Hence, applying the Landau-Miller algorithm [Landau and Miller 1985] we can compute all the prime factors of $\#\mathrm{Gal}\,(f)$ and $\#\mathrm{Gal}\,(\mu_{\Delta_m})$ (Theorem 3.3). Thus, if we can compute in polynomial time the primitive polynomials $\mu_{\Delta_i}(X)$ of the fields $\mathbb{Q}_{\Delta_i}$ for $1 \le i \le m$ then we will have a polynomial-time algorithm to verify the conditions of Proposition 6.2. The following theorem is due to Landau and Miller [Landau and Miller 1985] restated in a form suitable for our application. For completeness, we present a proof in our notation.

THEOREM 6.3 (LANDAU-MILLER). *Let $f(X) \in \mathbb{Q}[X]$ be irreducible, $G = \mathrm{Gal}\,(f)$ be its Galois group and Ω be the set of roots of f over the algebraic closure $\overline{\mathbb{Q}}$. Let $\Delta \subseteq \Omega$ be any G-block and $\alpha \in \Delta$. There is an algorithm that takes as input a polynomial $p_\Delta(X) \in \mathbb{Q}[X]$ such that $\mathbb{Q}_\Delta = \mathbb{Q}(p_\Delta(\alpha))$, runs in time polynomial in* $\mathrm{size}\,(f)$ *and* $\mathrm{size}\,(p_\Delta)$*, and for each G-block Σ such that Δ is a maximal block of Σ, computes a polynomial $p_\Sigma(X) \in \mathbb{Q}[X]$ such that $\mathbb{Q}_\Sigma = \mathbb{Q}(p_\Sigma(\alpha))$. Furthermore, the size of the computed polynomial* $\mathrm{size}\,(p_\Sigma)$ *is bounded by a polynomial in* $\mathrm{size}\,(f)$ *and is independent of* $\mathrm{size}\,(p_\Delta)$.

Remark 6.4. A couple of remarks are in order before we proceed to the proof. Notice that the algorithm for computing p_Σ takes as input polynomials f as well as p_Δ. However, the theorem stipulates that the *size* of the output polynomial p_Σ is polynomially bounded in just the $\mathrm{size}\,(f)$ and *not* on the other polynomial $\mathrm{size}\,(p_\Delta)$. This property of the algorithm is crucial because we will recursively apply this algorithm to a tower of blocks, where the tower length can be logarithmic in $\deg(f)$. So, if $\mathrm{size}\,(p_\Sigma)$ had been a polynomial in $\mathrm{size}\,(p_\Delta)$ the overall algorithm would have incurred a polynomial size growth at every level of the tower making it superpolynomial.

Another point about the algorithm is that the field $\mathbb{Q}(\alpha)$ is identified with the quotient $\mathbb{Q}[X]/f(X)$. Thus, elements of $\mathbb{Q}(\alpha)$ are polynomials in α with rational coefficients. The algorithm will work with such polynomials representing elements of $\mathbb{Q}(\alpha)$.

PROOF. Consider the Galois group G as a permutation group over the roots Ω. For $\Delta \subseteq \Omega$ let $T_\Delta(X)$ denote the polynomial

$$T_\Delta(X) = \prod_{\eta \in \Delta} (X - \eta).$$

CLAIM 6.5. *For the G-block containing α if the polynomial $T_\Delta(X)$ defined above is $\delta_0 + \ldots + \delta_r X^r$. Then field $\mathbb{Q}_\Delta$ is the field $\mathbb{Q}(\delta_0, \ldots, \delta_r)$. Here $\delta_i \in \mathbb{Q}(\alpha)$ are polynomials in α with coefficients in $\mathbb{Q}$.*

PROOF OF CLAIM. Let K be the field $\mathbb{Q}(\delta_0, \ldots, \delta_r)$ and let H be the Galois group $\mathrm{Gal}\,(\mathbb{Q}_f/K)$. For the claim it is sufficient to prove that $H = G_\Delta$. Consider any automorphism $\sigma \in G_\Delta$. Since σ permutes the roots of Δ among themselves $\sigma(T_\Delta(X)) = T_\Delta$. So, σ has to fix each of the coefficients δ_i of T_Δ and hence fixes K. Conversely, consider any automorphism τ of H and let Δ' be the block Δ^τ. Since τ fixes K we have $\tau(T_\Delta(X)) = T_{\Delta'}$. As Δ and Δ' are G-blocks, this can only happen if $\Delta' = \Delta$ for otherwise $T_\Delta(X)$ and $T_{\Delta'}(X)$ have no common roots. Therefore, $\tau \in G_\Delta$. $\square$

Given the coefficients of the polynomial T_Δ, notice that we can compute p_Δ in polynomial time by applying the primitive element theorem (see Lemma 2.4). To see this, observe that $\mathbb{Q}_\Delta$ is a subfield of $\mathbb{Q}_\alpha$. Hence $[\mathbb{Q}_\Delta : \mathbb{Q}] \leq \deg(f)$. Therefore, the algorithm given by the primitive element theorem for computing the coefficients of p_Δ is polynomial-time bounded. Further, since $T_\Delta(X)$ is a factor of the polynomial $f(X)$, by a well-known result of Mignotte [Mignotte 1974], each of the δ_i's have size bounded by a polynomial in size (f).

Claim 6.6. *Let Δ be a G-block containing α. The irreducible factor of f over $\mathbb{Q}_\Delta$ which has α as root is T_Δ. Let Σ be any G-block such that $\Sigma \supseteq \Delta$. If g is an irreducible factor of f over $\mathbb{Q}_\Delta$ then Σ contains a root of g if and only if it contains all the roots of g.*

Proof of claim. Let g be an irreducible factor of $f(X)$ over $\mathbb{Q}_\Delta$. The roots of g form a G_Δ-orbit of Ω. Conversely, for any G_Δ-orbit Ω' the polynomial $T_{\Omega'}(X)$ is an irreducible factor of $f(X)$ over $\mathbb{Q}_\Delta$. Hence the irreducible factor of f over $\mathbb{Q}_\Delta$ that has α as root is T_Δ.

For a G-block Σ containing Δ we have $G_\Sigma \geq G_\Delta$. Hence, any orbit of G_Δ is completely contained inside an orbit of G_Σ. As the roots of any irreducible factor $g(X)$ of $f(X)$ over $\mathbb{Q}_\Delta$ form an orbit of G_Δ, it is completely contained inside a G_Σ orbit. Hence, if one of the roots of g is in Σ, the orbit α^{G_Σ} of G_Σ, then all roots are in Σ. This proves the claim. $\square$

Let Δ be a G-block containing α and assume that we have already computed the polynomial p_Δ. Further, assume that f factors as $g_0 \ldots g_r$ over $\mathbb{Q}_\Delta = \mathbb{Q}(p_\Delta(\alpha))$ (which can be computed in polynomial time by Landau's factorisation algorithm [Landau 1985]). One of these factors say g_0 is T_Δ. Consider any G-block Σ such that Δ is a maximal G-subblock of Σ. There is a factor g_i such that Σ contains a root, and hence all the roots (Claim 6.6) of g_i. Let Σ_i be the smallest G-block containing Δ and all the roots of g_i. We give a polynomial-time algorithm to compute T_{Σ_i}. Theorem 6.3 then follows from this algorithm.

Claim 6.7. *Let Δ be a G-block containing α. Given a polynomial p_Δ such that $\mathbb{Q}_\Delta = \mathbb{Q}(p_\Delta(\alpha))$ as a subfield of $\mathbb{Q}(\alpha)$ and an irreducible factor g of f over $\mathbb{Q}_\Delta$ we can compute in polynomial time T_Σ as a polynomial in $\mathbb{Q}(\alpha)[Y]$, where Σ is the smallest G-block containing Δ and the roots of g.*

Proof of Claim. We are given $\mathbb{Q}_\Delta$ as a subfield of $\mathbb{Q}(\alpha)$. The coefficients of factors of f over $\mathbb{Q}_\Delta$ are polynomials in α. Let the factorisation of f over $\mathbb{Q}_\Delta$ be $f = g_0 \ldots g_r$, where $g_0 = T_\Delta$ and $g = g_1$. Denote the set of roots of g_i by Φ_i, for each i. Then Φ_i's are the orbits of G_Δ and by Claim 6.6, the polynomial T_Σ is precisely the product of g_i such that $\Phi_i \subseteq \Sigma$.

Let β denote a root of $g(X)$, and $\sigma \in \mathrm{Gal}\,(\mathbb{Q}_f/\mathbb{Q})$ be an automorphism such that σ maps α to β. Notice that σ is an isomorphism between the fields $\mathbb{Q}(\alpha)$ and $\mathbb{Q}(\beta)$. Let Σ be the smallest G-block containing Δ and Φ_1. From Theorem 4.3 and the Galois correspondence of blocks (Theorem 4.3) we know that G_Σ is generated by $G_\Delta \cup \{\sigma\}$.

If generators for G_Δ and the automorphism σ are known, then the block Σ can be computed by transitive closure of procedure as in Algorithm 1. The correctness of this algorithm follows directly from Claim 6.6.

Our goal is to get a polynomial-time algorithm for computing T_Σ from the above procedure that defines Σ. First, we compute the extension field $\mathbb{Q}(\alpha, \beta) = \mathbb{Q}(\gamma)$: we do this by first factoring f over $\mathbb{Q}(\alpha)$. Let h be an irreducible factor of g over $\mathbb{Q}(\alpha)$. Then $\mathbb{Q}(\alpha, \beta) = \mathbb{Q}(\alpha)[X]/h(X)$. As $[\mathbb{Q}(\alpha, \beta) : \mathbb{Q}] \leq n^2$, we can compute a primitive element γ in polynomial time.[3] Furthermore, in polynomial time we can find polynomials r_1 and r_2 such that $\alpha = r_1(\gamma)$ and $\beta = r_2(\gamma)$.

[3] Note that we need to invoke Remark 2.5 for this computation.

```
Let $S := \{\Delta, \Phi_1\}$
while new orbits get added to $S$ do
    Compute $S' := \{\Phi^\sigma \mid \Phi \in S\}$
    if $\Phi_j \cap \Phi^\sigma \neq \emptyset$ for some $\Phi^\sigma \in S'$ then include $\Phi_j$ in $S$;
end
Output $\bigcup\{\Phi \mid \Phi \in S\}$
```

Algorithm 1: Computing Σ

For all $1 \leq i \leq r$, let σ map the polynomials g_i in $\mathbb{Q}(\alpha)[X]$ to the polynomials g_i^σ in $\mathbb{Q}(\beta)[X]$, obtained by symbolically replacing α by β in each coefficient of g_i. In Algorithm 1, testing if $\Phi_j \cap \Phi_i^\sigma \neq \emptyset$ amounts to finding if $gcd(g_j, g_i^\sigma)$ is nontrivial. To make this gcd computation possible, we must express g_j and g_i over $\mathbb{Q}(\gamma)$, which we do by replacing α by $r_1(\gamma)$ and β by $r_2(\gamma)$. We can now give the algorithm for computing T_Σ.

```
Let $S := \{T_\Delta, g\}$
while new factors get included in $S$ do
    Compute $S' := \{h^\sigma \mid h \in S\}$
    for each factor $g_j$ and $h^\sigma \in S'$ do
        Express $g_j(X)$ and $h^\sigma(X)$ as polynomials over the field $\mathbb{Q}(\gamma)$.
        if $gcd(g_j, h^\sigma)$ is nontrivial then include $g_j$ in $S$;
    end
end
Output $T_\Sigma := T_\Delta \cdot \prod_{g_i \in S} g_i$
```

Algorithm 2: Computing T_Σ

It is clear that Algorithm 2 is polynomial-time bounded. The preceding discussion and the procedure for defining Σ imply that the algorithm correctly computes T_Σ. This proves Claim 6.7. □

The Algorithm

The complete nilpotence test is given in Algorithm 3. We show that the algorithm is correct and that its running time is polynomially bounded in its input size in the rest of the section.

PROPOSITION 6.8. *Algorithm 3 runs in time polynomial in* size(f).

PROOF. The Landau-Miller solvability test for $\mathrm{Gal}(\mathbb{Q}_f/\mathbb{Q})$ and the algorithm of Theorem 3.3 are polynomial time bounded [Landau and Miller 1985]. The nilpotence test first verifies that $\mathrm{Gal}(\mathbb{Q}_f/\mathbb{Q})$ is solvable by applying the Landau-Miller test. Now, since the field $\mathbb{Q}_{\Delta_m}$ is contained in $\mathbb{Q}_f$, its primitive polynomial $\mu_{\Delta_m}(X)$ will also split in $\mathbb{Q}_f$ implying that $\mathbb{Q}_{\mu_{\Delta_m}} \subset \mathbb{Q}_f$. Hence $\mathrm{Gal}\left(\mathbb{Q}_{\mu_{\Delta_m}}/\mathbb{Q}\right) = \mathrm{Gal}(\mu_{\Delta_m})$ is also a solvable group. Hence, in steps 1 and 7 we can apply the algorithm of Theorem 3.3 to compute all the prime factors of $\#\mathrm{Gal}(f)$ and $\#\mathrm{Gal}(\mu_{\Delta_m})$ in polynomial time.

Step 5 can be carried out in polynomial time by Theorem 6.3, and Step 6 can be carried out in polynomial time (applying the algorithmic version of the primitive element theorem as explained in Remark 2.5).

Clearly all other steps within the loop starting at line 4 can be carried out in polynomial time. Hence the overall algorithm is polynomial-time bounded. □

We now prove the correctness of the algorithm in the next two propositions.

PROPOSITION 6.9. *If $f(X)$ is an input irreducible polynomial of degree n such that* $\mathrm{Gal}(f)$ *is nilpotent then Algorithm 3 accepts f.*

```
  **Input:** A polynomial $f(X) \in \mathbb{Q}[X]$ of degree $n$
  **Output:** *Accept* if $\mathrm{Gal}(f)$ is nilpotent;*Reject* otherwise
  Verify that $f(X)$ is solvable using the Landau-Miller test.;
1 Compute the set $P$ of all the prime factors of $\#\mathrm{Gal}(f)$;
  Let $G \leq \mathrm{Sym}(\Omega)$ denote the Galois group of $f$, where $\Omega$ is the set of roots of $f$.
2 for every $p \in P$ do
3     if $p$ does not divide $n$ then Reject;
      Let $p^m$ be the highest power of $p$ dividing $n$.
      $\mathbb{Q}_{\Delta_0} := \mathbb{Q}(X)/f(X)$
4     for $i = 0$ to $m - 1$ do
5         By Theorem 6.3 compute $\mathbb{Q}_\Gamma$ for all minimal $G$-blocks $\Gamma$ containing $\Delta_i$.
6         Among the fields $\mathbb{Q}_\Gamma$ computed above check if there a field $K$ such that $\mathbb{Q}_{\Delta_i}/K$
            is a normal extension of degree $[\mathbb{Q}_{\Delta_i} : K] = p$.
          if no such field exists then Reject;
          else $\mathbb{Q}_{\Delta_{i+1}} := K$ ;
      end
      Let $\mu_{\Delta_m}(X)$ be the primitive polynomial for $\mathbb{Q}_{\Delta_m}$
7     if $p$ divides $\#\mathrm{Gal}(\mu_{\Delta_m})$ then Reject;
  end
  Accept
```

Algorithm 3: Nilpotence test

Proof. Let G be the Galois group $\mathrm{Gal}(f)$ and let Ω be the set of roots of f. Since f is of degree n, $\#\Omega = n$ and by Lemma 5.3 every prime factor of $\#G$ divides n. Therefore, algorithm never rejects f at step 3. Now, for the loop starting in line 4 we show that if G is nilpotent the algorithm always succeeds in finding a field K, from among the candidate fields $\mathbb{Q}_\Gamma$, in step 6. Notice that at the i^{th} iteration the block Δ_i is of cardinality p^i and $i < m$. Hence, by Theorem 5.9, Δ_i is contained in some p-Sylow block say Σ and there is a minimal G-block Δ containing Δ_i that has the following three properties:

(1) The index $[\Delta : \Delta_i]$ is p.
(2) The group G_{Δ_i} is a normal subgroup of G_Δ.
(3) The quotient G_Δ / G_{Δ_i} is cyclic of order p.

Consider the field $\mathbb{Q}_\Delta$. Since Δ is a minimal block that properly contains Δ_i, the field $\mathbb{Q}_\Delta$ is among the fields computed in Step 5. We claim that $\mathbb{Q}_\Delta$ is a suitable choice for K in step 6. The groups G_{Δ_i} and G_Δ are the Galois groups $\mathrm{Gal}(\mathbb{Q}_f/\mathbb{Q}_{\Delta_i})$ and $\mathrm{Gal}(\mathbb{Q}_f/\mathbb{Q}_\Delta)$, respectively. Hence, by Theorem 2.1, we have $\mathbb{Q}_{\Delta_i}/\mathbb{Q}_\Delta$ is a normal extension of degree $[\mathbb{Q}_{\Delta_i} : \mathbb{Q}_\Delta] = [G_\Delta : G_{\Delta_i}] = p$. As the algorithm goes over all minimal G-blocks Γ containing Δ_i, it will always succeed in finding a field K in step 6.

Finally, at the end of the loop, the index i becomes m and Δ_m is of order p^m. Since p^m is the highest power of p dividing $\#\Omega$, by Theorem 5.9 the block Δ_m is a p-Sylow block. By Proposition 6.1 $G/G^{\Delta_m} = \mathrm{Gal}(\mu_{\Delta_m})$. Further, by Lemma 5.5 the group G^{Δ_m} is the unique p-Sylow subgroup of the nilpotent group G, which implies that p does not divide $\#G/G^{\Delta_m}$. Hence the input passes the test in step 7. □

Proposition 6.10. *If Algorithm 3 accepts the input polynomial $f(X)$ then* $\mathrm{Gal}(f)$ *is nilpotent.*

Proof. Let G be the Galois group of f. We claim that if the algorithm accepts the input then for every prime p dividing $\#G$ we have a maximal chain of $\{\alpha\} = \Delta_0 \subset \ldots \subset \Delta_m$ with the following properties

(1) The index of the block $[\Delta_{i+1} : \Delta_i] = p$,

(2) The group G_{Δ_i} is a normal subgroup of $G_{\Delta_{i+1}}$ and
(3) The prime p does not divide G/G^{Δ_m}.

This is because in step 6 we have verified that $\mathbb{Q}_{\Delta_i}$ is a normal extension of $\mathbb{Q}_{\Delta_{i+1}}$ of degree p. Hence by the fundamental theorem of Galois theory their Galois groups G_{Δ_i} and $G_{\Delta_{i+1}}$ are such that G_{Δ_i} is a normal subgroup of $G_{\Delta_{i+1}}$ and $[G_{\Delta_{i+1}} : G_{\Delta_i}] = [\Delta_{i+1} : \Delta_i] = p$. Furthermore, in step 7 we have verified that p does not divide the order of $\mathrm{Gal}\,(\mu_{\Delta_m}) = G/G^{\Delta_m}$. Therefore, G satisfies all the properties of Theorem 5.10 and hence is nilpotent. □

Propositions 6.8, 6.9 and 6.10 together show the following.

THEOREM 6.11. *There is a deterministic polynomial-time algorithm that takes as input $f(X)$ over $\mathbb{Q}$ and decides whether the Galois group of f is nilpotent.*

7. COMPUTING SYLOW POLYNOMIALS

In the last two sections we saw that Sylow subgroups play a crucial role in the nilpotence testing algorithm. In this section we explore whether any further information regarding Sylow subgroups of nilpotent Galois groups can be computed. In this context we make the following definition.

Definition 7.1 (*Sylow polynomials*). Let $f(X)$ be any polynomial over $\mathbb{Q}$ with nilpotent Galois group G. Let p be a prime that divides the order of G. By a p-Sylow polynomial we mean a polynomial $g(X)$ over $\mathbb{Q}$ such that $g(X)$ splits in the splitting field $\mathbb{Q}_f$ of f and the $\mathrm{Gal}\,(g)$ is (isomorphic to) the p-Sylow subgroup G_p of G.

In this section we describe a polynomial-time algorithm that, given as input a polynomial $f(X) \in \mathbb{Q}[X]$ with nilpotent Galois group G, computes a p-Sylow polynomial for each prime factor p of $\#G$. An immediate consequence is that for polynomials $f(X)$ with nilpotent Galois group there are Sylow polynomials of polynomially bounded degree.

In the following lemma we show that for this problem it suffices to consider irreducible polynomials $f(X)$.

LEMMA 7.2. *Let $f(X) \in \mathbb{Q}[X]$ be a polynomial with nilpotent Galois group and let $f_1, f_2, \ldots, f_k$ be all distinct irreducible factors of $f(X)$. For each i, let g_i be a p-Sylow polynomial for f_i. Then the product polynomial $g_1 g_2 \cdots g_k$ is a p-Sylow polynomial for $f(X)$.*

PROOF. Let $g(X) = g_1(X)g_2(X) \cdots g_k(X)$. Since g_i is a p-Sylow polynomial of f_i, by definition we g_i splits in the field $\mathbb{Q}_{f_i}$, and its Galois group $\mathrm{Gal}\,(g_i)$ is isomorphic to the unique p-Sylow subgroup $G_p^{(i)}$ of $G^{(i)} = \mathrm{Gal}\,(f_i)$. Hence the Galois group $\mathrm{Gal}\,(\mathbb{Q}_{f_i}/\mathbb{Q}_{g_i})$ is isomorphic to the quotient group $G^{(i)}/G_p^{(i)}$. Therefore, p does not divide the order of the Galois group $\mathrm{Gal}\,(\mathbb{Q}_{f_i}/\mathbb{Q}_{g_i})$.

For each i we have $\mathbb{Q}_{g_i} \subseteq \mathbb{Q}_{f_i} \subseteq \mathbb{Q}_f$. Let $\overline{\mathbb{Q}}$ denote the algebraic closure of $\mathbb{Q}$. Observe that $\mathbb{Q}_f$ is the smallest subfield of $\overline{\mathbb{Q}}$ containing $\mathbb{Q}_{f_i}$ for each f_i. Likewise, $\mathbb{Q}_g$ is the smallest subfield of $\overline{Q}$ containing each $\mathbb{Q}_{g_i}$. It follows that $\mathbb{Q}_g$ is a subfield of $\mathbb{Q}_f$.

Furthermore, we can observe that every $\sigma \in \mathrm{Gal}\,(\mathbb{Q}_g/\mathbb{Q})$ must map g_i to g_i for each i. Hence, for each i, σ restricted to $\mathbb{Q}_{g_i}$ is in $\mathrm{Gal}\,(\mathbb{Q}_{g_i}/\mathbb{Q})$. Clearly, σ is nontrivial if and only if it is nontrivial on some $\mathbb{Q}_{g_i}$. Therefore, since each $\mathrm{Gal}\,(\mathbb{Q}_{g_i}/\mathbb{Q})$ is a p-group it follows that $\mathrm{Gal}\,(\mathbb{Q}_g/\mathbb{Q})$ is also a p-group.

Similarly, consider an element $\pi \in \mathrm{Gal}\,(\mathbb{Q}_f/\mathbb{Q}_g)$. We can see that for each i, π restricted to $\mathbb{Q}_{f_i}$ is in $\mathrm{Gal}\,(\mathbb{Q}_{f_i}/\mathbb{Q}_{g_i})$ and π is nontrivial only if it is nontrivial on some $\mathbb{Q}_{f_i}$.

As a result the Galois group $\mathrm{Gal}\,(\mathbb{Q}_f/\mathbb{Q}_g)$ is isomorphic to a subgroup of the product group $\prod_i \mathrm{Gal}\,(\mathbb{Q}_{f_i}/\mathbb{Q}_{g_i})$. Hence, p is not a factor of $\#\mathrm{Gal}\,(\mathbb{Q}_f/\mathbb{Q}_g)$ as it is not a factor of $\#\mathrm{Gal}\,(\mathbb{Q}_{f_i}/\mathbb{Q}_{g_i})$ for $1 \leq i \leq k$. It follows that $\mathrm{Gal}\,(\mathbb{Q}_g/\mathbb{Q})$ is isomorphic to the p-Sylow subgroup of $\mathrm{Gal}\,(\mathbb{Q}_f/\mathbb{Q})$. Hence g is a p-Sylow polynomial for f. □

We now consider the problem of computing a p-Sylow polynomial for an irreducible polynomial $f(X)$ with nilpotent Galois group. To this end we generalise the notion of Sylow blocks (Definition 5.1).

Definition 7.3 (*Generalised Sylow block*). Let G be a transitive nilpotent permutation group on Ω and let $\{p_1, p_2, \ldots, p_k\}$ be the set of all prime factors of $\#G$. For $\emptyset \subset P \subseteq \{p_1, p_2, \ldots, p_k\}$, a subset $\Sigma \subseteq \Omega$ is called a *P-Sylow block* if it is an orbit of $G_P = \prod_{p \in P} G_p$.

Since G_P is a normal subgroup of G, each orbit of G_P is a block of G.

The next three results state some properties of P-Sylow blocks of transitive nilpotent groups that we require for the algorithm. These are generalisations of Lemma 5.3, Lemma 5.5 and Theorem 5.9 respectively. We give brief proofs for these since they are on the same lines as their counterparts in Section **??**. As a matter of fact, all results in Section 5 for Sylow blocks have straightforward generalisations to P-Sylow blocks.

LEMMA 7.4. *Let $G \leq \mathrm{Sym}(\Omega)$ be a transitive nilpotent permutation group and let P be any set of primes that divide the order of G. Let m_p denote the highest power of p that divides $\#\Omega$. Then any P-Sylow block Σ has cardinality $\prod_{p \in P} p^{m_p}$.*

PROOF. By Theorem 4.3 $[\Omega : \Sigma] = [G : G_\Sigma]$. Recall that $G_\Sigma = \{g \in G \mid \Sigma^g = \Sigma\}$. Since Σ is a G_P orbit it follows that G_P is a subgroup of G_Σ. Hence for every prime $p \in P$, p does not divide $\#G/\#G_\Sigma$, and hence p does not divide $\#\Omega/\#\Sigma$ for each $p \in P$. By the nilpotence of G, each $p \in P$ divides $\#\Omega$. Therefore, p^{m_p} divides $\#\Omega$ for each $p \in P$.

On the other hand, since Σ is a G_P-orbit, G_P is transitive on Σ. Hence, by the Orbit-Stabilizer formula (Theorem 4.1) $\#\Sigma$ divides $\#G_P$ which means $\#\Sigma$ has no prime factors other than from P. Putting it together, it follows that Σ has cardinality $\prod_{p \in P} p^{m_p}$. □

LEMMA 7.5. *Let $G \leq \mathrm{Sym}(\Omega)$ be a transitive permutation group and let P be any set of primes that divide the order of G. Let Σ be any P-Sylow block of G then G^Σ is the product $\prod_{p \in P} G_p$.*

PROOF. As observed in the previous lemma, G_P is a subgroup of G_Σ. Since Σ is a block for G, consider the block system generated by G-action on Σ and let Σ' be any other block in this system. For some $g \in G$ we have

$$G_{\Sigma'} \;=\; g G_\Sigma g^{-1}.$$

Since G_P is a normal subgroup of G, $g G_P g^{-1} = G_p$, and hence G_P is a subgroup of $G_{\Sigma'}$ for each Σ' in the block system which implies G_P is a subgroup of G^Σ.

Suppose that $G_P \neq G^\Sigma$. Then some prime $p \notin P$ divides $\#G^\Sigma$. Let G_p^Σ be the p-Sylow subgroup of G^Σ (which is unique and hence a normal subgroup since G^Σ is also nilpotent). Suppose G_p^Σ has nontrivial action on some block Σ' in the block system. If $O \subset \Sigma'$ is a nontrivial G_p^Σ-orbit then it is a G^Σ-block contained in Σ'. Further, G^Σ is transitive on Σ' since $G_P \leq G^\Sigma$. Hence $|O|$ divides $|\Sigma'|$ which is impossible since $|O|$ is a power of p and $|\Sigma'|$ does not have p as factor. This is a contradiction. Hence $G_P = G^\Sigma$. □

THEOREM 7.6. *Let G be any transitive permutation group on Ω and let P be a set of primes that divide the order of G. Let Γ be any G-block such that each prime that divides $\#\Gamma$ is in P. Then there is a P-Sylow block Σ of G such that $\Gamma \subset \Sigma$. Furthermore, for a prime $p \in P$ if p divides $\frac{\#\Omega}{\#\Gamma}$ then we have a G-block Γ' such that $\Gamma \subset \Gamma' \subseteq \Sigma$ such that*

(*1*) *The index $[\Gamma' : \Gamma]$ is p.*
(*2*) *The group G_Γ is a normal subgroup of $G_{\Gamma'}$.*
(*3*) *The quotient group $G_{\Gamma'}/G_\Gamma$ is a cyclic group of order p.*

PROOF. Let Γ be any G-block and $\alpha \in \Gamma$. Let $\Sigma = \alpha^{G_P}$ which, by definition, is a P-Sylow block. We claim that $\Gamma \subseteq \Sigma$. Let $\{p_1, p_2, \ldots, p_k\}$ be the set of all distinct prime factors of $\#G$ and let $\overline{P}$ denote the complement of P in this set.

Now consider the group G_Γ, which is transitive on Γ by Proposition 4.2. Since G_Γ, being nilpotent, is the direct product of its Sylow subgroups, we can write

$$G_\Gamma \;=\; G_{\Gamma,P} \times G_{\Gamma,\overline{P}},$$

where $G_{\Gamma,P}$ is the product of the p-Sylow subgroups of G_Γ for $p \in P$ and $G_{\Gamma,\overline{P}}$ is similarly defined for $\overline{P}$. Now, every element of $G_{\Gamma,\overline{P}}$ must pointwise fix Γ because $G_{\Gamma,\overline{P}}$ is a normal subgroup of G_Γ and a nontrivial orbit of $G_{\Gamma,\overline{P}}$ will have size whose prime factors are all from $\overline{P}$ on the one hand, and on the other hand the orbit size must divide $\#\Gamma$. It follows that $G_{\Gamma,P}$ is transitive on Γ and hence $\alpha^{G_{\Gamma,P}} = \Gamma$. Since $G_{\Gamma,P}$ is contained in G_P it follows that $\Gamma \subseteq \Sigma$.

Suppose $p \in P$ divides $\#\Omega/\#\Gamma$. To prove the three parts of the theorem, we will consider the action of the group G on the block system $\mathcal{B}(\Omega/\Gamma)$. Since G is nilpotent and transitive on $\mathcal{B}(\Omega/\Gamma)$ and p divides $\#\Omega/\#\Gamma$ we can apply Theorem 5.9. The block Δ in Theorem 5.9 is set to be the singleton set $\{\Gamma\}$. By Theorem 5.9 there is a p-Sylow block $\Sigma' = \{\Gamma_1 = \Gamma, \Gamma_2, \ldots, \Gamma_t\}$ containing Γ and a minimal G-block $\Delta' \subset \Sigma'$ of size p. Without loss of generality, let

$$\Delta' = \{\Gamma_1 = \Gamma, \Gamma_2, \ldots, \Gamma_p\}.$$

Let $\Gamma' = \bigcup_{i=1}^{p} \Gamma_i$. It is easy to verify that all the three conditions in the statement follow from the corresponding conditions for Δ' and Δ in Theorem 5.9. □

Let $\{p_1, p_2, \ldots, p_k\}$ be the set of all distinct prime factors of $\#\mathrm{Gal}\,(f)$. In order to compute the p_i-Sylow polynomial we set $P_i = \{p_1, p_2, \ldots, p_k\} \setminus \{p_i\}$ and compute a tower of blocks $\{\alpha\} = \Delta_0 \subset \Delta_1 \subset \ldots \subset \Delta_m$ where Δ_m is the P_i-Sylow block containing the point α. By computing the blocks Δ_i we mean, as in Section 6, that we compute a primitive polynomial $\mu_{\Delta_i}(X)$ for the field $\mathbb{Q}_{\Delta_i}$, $1 \leq i \leq m$. Since the Galois group of μ_{Δ_i} is G/G^{Δ_i} (by Proposition 6.1), and $G^{\Delta_m} = G_{P_i}$ it follows that the Galois group of μ_{Δ_m} is $G/G^{\Delta_m} = G/G_{P_i}$ which is isomorphic to G_{p_i}, the p_i-Sylow subgroup of G. Hence μ_{Δ_m} is a p_i-Sylow polynomial for f.

We now give the complete algorithm for computing the p-Sylow polynomial for an irreducible polynomial $f(X)$ with nilpotent Galois group.

Clearly Algorithm 4 runs in time polynomial in size$\,(f)$. By Theorem 7.6 it follows that step 4 is always possible. Therefore, it follows from Lemma 7.4 that at the end of the two loops we have a primitive polynomial $\mu_{\Delta_m}(X)$ for $\mathbb{Q}_{\Delta_m}$ where Δ_m is a P_i-Sylow block of G. The Galois group $\mathrm{Gal}\,(\mu_{\Delta_m})$ is G/G^{Δ_m} by Proposition 6.1. Hence the Galois group of $\mu_{\Delta_m}(X)$ is the p_i-Sylow subgroup of G as claimed. Algorithm 4 thus computes the p_i-Sylow polynomial for the irreducible polynomial $f(X)$. Together with Lemma 7.2 we have the following theorem.

THEOREM 7.7. *There is a deterministic polynomial-time algorithm that given a polynomial $f(X)$ with nilpotent Galois group and any prime p dividing the order of the Galois group* Gal$\,(f)$*, computes a polynomial $g(X)$ such that $g(X)$ splits in $\mathbb{Q}_f$ and the Galois group of $g(X)$ is isomorphic to the p-Sylow subgroup of* Gal$\,(f)$.

In fact the same ideas yield a more general observation by modifying the Algorithm 4 to work with any arbitrary set P of prime factors of $\#\mathrm{Gal}\,(f)$.

THEOREM 7.8. *There is a deterministic polynomial-time algorithm that given a polynomial $f(X)$ with nilpotent Galois group and any subset P of primes dividing the order of the Galois group* Gal$\,(f)$*, computes a polynomial $g(X)$ such that $g(X)$ splits in $\mathbb{Q}_f$ and the*

```
Input: An irreducible polynomial $f(X) \in \mathbb{Q}[X]$ of degree $n$ with nilpotent Galois
       group and a prime factor $p_i$ of $n$.
Output: A $p_i$-Sylow polynomial of $f$
Let $\{p_1, p_2, \ldots, p_k\}$ be the set of all prime factors of $\#\mathrm{Gal}(f)$ (these are exactly the
  prime factors of $n$ by Lemma 5.3).
Let $\mathbb{Q}_{\Delta_0} = \mathbb{Q}(X)/f(X)$ and $\mu_{\Delta_0}(X) = f(X)$.
Let $G$ denote the Galois group of $f$.
$r := 0$
$\mathbb{Q}_{\Delta_0} := \mathbb{Q}(X)/f(X)$

for $j \in \{1, \ldots, i-1, i+1, \ldots, k\}$ do
    Let $p_j^{m_j}$ be the highest power of $p_j$ dividing $n$.
    for $\ell = 1$ to $m_j$ do
        Using Theorem 6.3 compute $\mathbb{Q}_\Gamma$ for all minimal $G$-blocks $\Gamma$ containing $\Delta_r$.
        Among the fields $\mathbb{Q}_\Gamma$ computed above find a field $\mathbb{Q}_{\Delta_{r+1}}$ such that $\mathbb{Q}_{\Delta_r}/\mathbb{Q}_{\Delta_{r+1}}$
          is a normal extension of degree $p_j$.
        $r := r + 1$
    end
end
Let $\mu_{\Delta_m}(X)$ be the primitive polynomial for $\mathbb{Q}_{\Delta_m}$, where $m = \sum_{j \neq i} m_j$
return $\mu_{\Delta_m}(X)$
```

Algorithm 4: Computing a p_i-Sylow polynomial

Galois group of $g(X)$ is (isomorphic to) the subgroup $G_P = \prod_{p \in P} G_p$ where G_p denotes the unique p-Sylow subgroup of $\mathrm{Gal}(f)$.

8. CONCLUDING REMARKS

Computing the Galois group of a polynomial $f(X) \in \mathbb{Q}[X]$ efficiently remains a challenging open problem. However, it is possible to test certain properties like commutativity and solvability efficiently. We have added nilpotence testing to this list. In this context, an intriguing problem is whether we can efficiently test if the Galois group of $f(X)$ is supersolvable (refer Hall's text [Hall Jr. 1959, Chapter 10] for a definition). Supersolvable groups are a proper subclass of solvable groups and contain nilpotent groups. It is not clear if we can adapt either the Landau-Miller solvability test [Landau and Miller 1985] or our nilpotence test to an efficient supersovablility test. It would be interesting to even give a conditionally efficient algorithm, e.g. assuming the generalised Riemann hypothesis.

Finally, we note that our nilpotent test can be generalised to obtain a polynomial-time algorithm to test if the Galois group of a polynomial $f(X) \in K[X]$ is nilpotent, where K is a number field presented by giving a primitive polynomial $\mu(X)$ of K over $\mathbb{Q}$, and the running time is polynomially bounded in $\mathrm{size}(f)$ and $\mathrm{size}(\mu)$. This generalised nilpotence test requires some standard polynomial-time algorithms like factoring of univariate polynomials and gcd computations over K which are already known [Landau 1985].

Acknowledgments. We are indebted to the referees for their meticulous refereeing with many important suggestions and corrections.

REFERENCES

Ákos Seress. 2003. *Permutation group algorithms.* Number 152 in Cambridge Tracts in Mathematics. Cambridge University Press.

Arvind, V. and Kurur, P. P. 2006. A polynomial time nilpotence test for galois groups and related results. In *31st International Symposium on Mathematical Foundations of Computer Science*. Lecture Notes in Computer Science 4162, Springer Verlag, 134–145.

Cohen, H. 1993. *A course in Computational Number Theory*. Springer-Verlag, Berlin, Heidelberg, New York.

Fernandez-Ferreiros, P. and Gomez-Molleda, M. A. 2003. Deciding the nilpotency of the Galois group by computing elements in the centre. *Mathematics of Computation 73,* 248.

Galois, É. 1830a. Analyse d'un mémoire sur la résolution algébrique des équations. *Bulletin des Sciences mathématiques XIII*, 271.

Galois, É. 1830b. Note sur la résolution des équations numériques. *Bulletin des Sciences mathématiques XIII*, 413.

Geissler, K. and Klüners, J. 2000. Galois group computation for rational polynomials. *Journal of Symbolic Computation 30*, 653–674.

Hall Jr., M. 1959. *The Theory of Groups* first Ed. The Macmillan Company, New York.

Jerrum, M. 1986. A compact representation for permutation groups. *Journal of Algorithms 7,* 1, 60–78.

Kurur, P. P. 2006. Complexity upper bounds using permutation group theory. Ph.D. thesis, Institute of Mathematical Sciences, Chennai, India.

Landau, S. 1984. Polynomial time algorithms for Galois groups. In *EUROSAM 84 Proceedings of International Symposium on Symbolic and Algebraic Computation*, J. Fitch, Ed. Lecture Notes in Computer Sciences Series, vol. 174. Springer, 225–236.

Landau, S. 1985. Factoring polynomials over algebraic number fields. *SIAM Journal of Computing 14*, 184–195.

Landau, S. and Miller, G. L. 1985. Solvability by radicals is in polynomial time. *Journal of Computer and System Sciences 30*, 179–208.

Lang, S. 1999. *Algebra* third Ed. Addison-Wesley Publishing Company, Inc.

Lenstra, A. K., Lenstra Jr. H. W., and Lovász, L. 1982. Factoring polynomials with rational coefficients. *Mathematische Annalen 261*, 515–534.

Lenstra Jr., H. W. 1992. Algorithms in algebraic number theory. *Bulletin of the American Mathematical Society 26,* 2, 211–244.

Luks, E. M. 1982. Isomorphism of graphs of bounded valence can be tested in polynomial time. *Journal of Computer and System Sciences 25,* 1, 42–65.

Luks, E. M. 1993. Permutation groups and polynomial time computations. *DIMACS Series in Discrete Mathematics and Theoretical Computer Science 11*, 139–175.

Mattman, T. and McKay, J. 1985. Computation of Galois groups over function fields. *Mathematics of Computation 66*, 823–831.

Matzat, B., McKay, J., and Yokoyama., K. 2000. Special issue on algorithmic methods in galois theory. *Journal of Symbolic Computation 30*.

Mignotte, M. 1974. An inequality about factors of polynomials. *Mathematics of Computation 28,* 128, 1153–1157.

Shoup, V. 1999. Efficient computation of minimal polynomials in algebraic extension of finite fields. In *International Symposium on Symbolic and Algebraic Computation*. 53–58.

Soicher, L. and McKay, J. 1985. Computing Galois groups over rationals. *Journal of Number Theory 20*, 273–281.

Stauduhar, R. P. 1973. The determination of Galois groups. *Mathematics of Computation 27*, 981–996.

Tschebotaröw, N. and Schwerdtfeger, N. 1950. Grundzüge der galoischen theorie. In *Groningen, Djakarta*. Noordhoff.

van der Waerden, B. L. 1991. *Algebra* Seventh Ed. Vol. I. Springer-Verlag.

Wielandt, H. 1964. *Finite Permutation Groups*. Academic Press, New York.