MATHEMATICS OF COMPUTATION
Volume 66, Number 218, April 1997, Pages 779–786
S 0025-5718(97)00839-9

COMPUTING IRREDUCIBLE REPRESENTATIONS OF SUPERSOLVABLE GROUPS OVER SMALL FINITE FIELDS

A. OMRANI AND A. SHOKROLLAHI

ABSTRACT. We present an algorithm to compute a full set of irreducible representations of a supersolvable group G over a finite field K, $\mathrm{char}K \nmid |G|$, which is not assumed to be a splitting field of G. The main subroutines of our algorithm are a modification of the algorithm of Baum and Clausen (Math. Comp. **63** (1994), 351–359) to obtain information on algebraically conjugate representations, and an effective version of Speiser's generalization of Hilbert's Theorem 90 stating that $H^1(\mathrm{Gal}(L/K), \mathrm{GL}(n, L))$ vanishes for all $n \geq 1$.

1. INTRODUCTION AND MAIN RESULTS

Recently Baum and Clausen [1] published an efficient algorithm for computing the absolutely irreducible representations of a supersolvable group G given in pc-presentation. The matrix representations their algorithm computes are adapted to a chief series $\mathcal{T} := (G = G_n > G_{n-1} > \cdots > G_0 = \{1\})$, i.e., any such representation D satisfies the following conditions: (1) the restriction $D \downarrow G_j$ of D to G_j is *equal* to a direct sum of irreducible matrix representations of G_j, and (2) equivalent irreducible constituents of $D \downarrow G_j$ are *equal.* The algorithm traverses the chief series $\mathcal{T}$ bottom-up and constructs in each step j among other data a complete set of inequivalent absolutely irreducible representations of G_j. These representations are almost unique: if L is a field containing a primitive eth root of unity, e being the exponent of G, and D and Δ are two equivalent irreducible $\mathcal{T}$-adapted representations of LG of degree d, say, then the *intertwining space*

$$\mathrm{Int}(D, \Delta) := \{X \in L^{d \times d} \mid \forall\, g \in G: \quad XD(g) = \Delta(g)X\}$$

is generated over L by a *monomial* matrix (see [2, Theorem 7.4]).

Now let K be a finite field, G be a supersolvable group such that $\mathrm{char}K \nmid |G|$, $\mathcal{T}$ be a chief series of G, and L be a finite extension of K which contains a primitive eth root of unity. The Galois group $\mathrm{Gal}(L/K)$ acts on the irreducible matrix representations of LG in a straightforward manner. In Section 2 we shall modify the algorithm of Baum and Clausen by collecting at each step information about the $\mathrm{Gal}(L/K)$-orbits of the representations constructed. We then employ the information obtained at level n to compute realizations of direct sums of these representations over the field K. By a realization of a matrix representation D of LG over K we mean a matrix $T \in \mathrm{GL}(d, L)$, d being the degree of D, such

Received by the editor May 23, 1995 and, in revised form, November 10, 1995 and May 1, 1996.

1991 *Mathematics Subject Classification.* Primary 20C15, 11R34, 20D15, 11T99.

Key words and phrases. Computational representation theory, Galois cohomology, p-groups, finite fields.

that $T^{-1}D(g)T$ has entries in K for all $g \in G$. Not every representation has a realization over K. Even more, if K is a prime field, χ denotes the character of D, and $K(\chi) := K(\chi(g) \mid g \in G)$ its character field, then D cannot have a realization over a proper subfield of $K(\chi)$. The question whether an absolutely irreducible representation D of G has a realization over $K(\chi)$ is hard to answer in general, i.e., for arbitrary K and arbitrary G. (This amounts to the question whether the Schur-index of the character of D equals 1, see [3, Kapitel V, §14].) It is, however, well known that for finite fields and arbitrary finite groups the question has an affirmative answer [3, Kapitel V, Satz 14.10].

In theory we thus know that any irreducible matrix representation D of LG has a realization over its character field. How can we compute such a realization? Let M be a subfield of L of index ℓ, and β be the Frobenius automorphism of L/M. If D is an irreducible representation of LG of degree d, then so is D^β, where $D^\beta(g) := D(g)^\beta$ for all $g \in G$. If M is the character field of D, then D is equivalent to D^β, hence $\mathrm{Int}(D, D^\beta)$ is generated by an invertible matrix S. A generalization of Hilbert's Theorem 90 due to Speiser [7] states that the first cohomology $H^1(\langle\beta\rangle, \mathrm{GL}(d, L))$ is trivial. (This is a modern interpretation of Speiser's result; see also [6, Chapter X, §1].) Hence, for $S \in \mathrm{GL}(d, L)$ there exists $T \in \mathrm{GL}(d, L)$ such that $T^{-1}T^\beta = S$ if and only if the norm $S^{\beta^{\ell-1}} \cdots S^\beta S$ of S equals the $n \times n$-identity matrix I_n. Such a matrix T will give the desired realization of D over its character field M. In our applications, S is a *monomial* matrix and this allows to compute T from S efficiently, see Section 3.

Now we are almost done. Namely, we may suppose that D is an absolutely irreducible representation of G with character χ such that $D(g)$ has entries in $K(\chi)$. Let σ be the Frobenius automorphism of $K(\chi)/K$. Then the *trace* of D *over* K defined as $\mathrm{Tr}_K(D) := D \oplus D^\sigma \oplus \cdots \oplus D^{\sigma^{m-1}}$, $m := [K(\chi) \colon K]$, has character field equal to K and a realization of $\mathrm{Tr}_K(D)$ over K can be computed easily, see Section 3. Furthermore, $\mathrm{Tr}_K(D)$ is an irreducible KG-representation (since any of its irreducible constituents over K has to be invariant under σ); conversely, any irreducible KG-representation is the trace over K of some irreducible MG-representation, where M is a splitting field of the representation in question containing K. (For these and related facts see [4, Chapter VII, §1].) To obtain the irreducible representations of KG we first compute a set $\mathcal{F}'$ of representatives of $\mathrm{Gal}(L/K)$-orbits of irreducible representations of LG, and for each such representation a realization of its trace over K. Starting from a pc-presentation of G and the chief series $\mathcal{T}$ induced by that, the first two steps of our algorithm are as follows:

Step 1. We first modify the algorithm of Baum and Clausen to compute a full set $\mathcal{F}$ of pairwise inequivalent irreducible monomial and $\mathcal{T}$-adapted representations of LG, where L is a field extension of K containing a primitive eth root of unity, and a permutation γ of $\mathcal{F}$ such that F^α is equivalent to γF: $F^\alpha \sim \gamma F$. Here α is the Frobenius automorphism of L/K. We then compute a full set $\mathcal{F}'$ of representatives of $\mathrm{Gal}(L/K)$-orbits of $\mathcal{F}$ and for each $F \in \mathcal{F}$ the degree of the character field of F over K.

Step 2. For each $F \in \mathcal{F}'$ we compute a realization T_F of F over its character field and then a realization of the trace of $T_F^{-1}FT_F$ over K.

Similar to the algorithm of Baum and Clausen, the arithmetic we use in these two steps consists just of symbolic computation in $L^\times$, where L is a field extension of K containing an eth root of unity. More precisely, we represent nonzero elements of L as integers i with $0 \le i < |L| - 1$, where i corresponds to the element ω^i and ω is a fixed generator of $L^\times$. This representation of L allows to solve efficiently equations of the form $\mathrm{N}(x) = \alpha$ or $x(x^\sigma)^{-1} = \alpha$, where $\alpha \in L$, N is the norm of L relative to a subfield M, and σ is the Frobenius automorphism of L/M. We shall need solutions to these kinds of equations in the second step of our algorithm. Moreover, as we will need primitive elements for subfields of L, this representation of L allows us to compute in advance these generators and store them in a list Ω.

The final step of the algorithm computes the KG-representations from the already computed realizations. This step requires matrix multiplication over L, and symbolic computation in $L^\times$ does not suffice for this purpose. Strategies to solve this problem are discussed in the last section.

Many thanks go to an anonymous referee for important comments and to Michael Clausen for communicating to us the problem discussed in this paper and for many stimulating discussions.

2. Irreducible LG-modules and $\mathrm{Gal}(L/K)$-orbits

The first step of our algorithm takes as input a supersolvable group G in pc-presentation and a finite extension L of K containing a primitive eth root of unity; it outputs a list $\mathcal{F}$ of pairwise inequivalent irreducible representations of LG and a permutation γ of $\mathcal{F}$ such that $F^\alpha \sim \gamma F$, α being the Frobenius automorphism of L/K.

For the rest of this section we set $\mathcal{T}_i := (G_i > G_{i-1} > \cdots > G_0 = \{1\})$ for $1 \le i \le n$. In particular, $\mathcal{T} = \mathcal{T}_n$. We call a matrix e-monomial if it is monomial and its nonzero entries are eth roots of unity. An LG representation F is called e-monomial if $F(g)$ is e-monomial for any $g \in G$.

The algorithm of Baum and Clausen in [1] computes the list $\mathcal{F}$; we modify this algorithm to obtain additional information on the orbits of $\mathcal{F}$ under the action of the Galois group of L/K; this information is encoded as the permutation γ.

Our algorithm works bottom up along $\mathcal{T}$. At level i, $1 \le i \le n$, it takes the following input:

(1) $\mathcal{F}$, a full set of inequivalent irreducible e-monomial representations of LG_{i-1} such that $\bigoplus_{F\in\mathcal{F}} F$ is $\mathcal{T}_{i-1}$-adapted;
(2) For every $i-1 < j \le n$ a permutation π_j of $\mathcal{F}$ such that $F^{g_j} \sim \pi_j F$ for all $F \in \mathcal{F}$ as well as e-monomial matrices $X_{j,F} \in \mathrm{Int}(F^{g_j}, \pi_j F)$, $F \in \mathcal{F}$;
(3) A permutation γ of $\mathcal{F}$ such that $F^\alpha \sim \gamma F$, as well as e-monomial matrices $M_F \in \mathrm{Int}(F^\alpha, \gamma F)$, $F \in \mathcal{F}$;

and computes the following output:

(1) $\mathcal{D}$, a full set of inequivalent irreducible e-monomial representations of LG_i such that $\bigoplus_{D\in\mathcal{D}} D$ is $\mathcal{T}_i$-adapted;
(2) For every $i < j \le n$ a permutation τ_j of $\mathcal{D}$ such that $D^{g_j} \sim \tau_j D$ for all $D \in \mathcal{D}$ as well as e-monomial matrices $Y_{j,D} \in \mathrm{Int}(D^{g_j}, \tau_j D)$, $D \in \mathcal{D}$.
(3) A permutation δ of $\mathcal{D}$ such that $D^\alpha \sim \delta D$, as well as e-monomial matrices $N_D \in \mathrm{Int}(D^\alpha, \delta D)$, $D \in \mathcal{D}$;

Outputs (1) and (2) are computed in exactly the same way as in the algorithm of Baum and Clausen [1]. Therefore, we only discuss the computation of Output (3) and assume that we have already performed the two phases of the algorithm in [1]. Note that during the construction at level i in Phase 1 there is built a bipartite graph in which $F \in \mathcal{F}$ and $D \in \mathcal{D}$ are linked if and only if F is a constituent of $D \downarrow G_{i-1}$. We will need this information to compute δ and N_D. For this we proceed in a similar way as does Phase 2 of the Baum-Clausen algorithm. Let $F \in \mathcal{F}$ and $p := [G_i \colon G_{i-1}]$. We distinguish two cases.

Case 1. Suppose that $\pi_i F = F$, i.e., $F^{g_i} \sim F$. Since $(F^{g_i})^\alpha = (F^\alpha)^{g_i}$, we obtain

$$(\gamma F)^{g_i} \sim (F^\alpha)^{g_i} = (F^{g_i})^\alpha \sim F^\alpha \sim \gamma F.$$

We already know p extensions $D_0, \dots, D_{p-1}$ of F and p extensions $\Delta_0, \dots, \Delta_{p-1}$ of γF. For $0 \le k < p$ we have

$$D_k^\alpha \downarrow G_{i-1} = (D_k \downarrow G_{i-1})^\alpha = F^\alpha \sim \gamma F,$$

hence D_k^α is equivalent to one of the representations $\Delta_0, \dots, \Delta_{p-1}$. Thus there exists a permutation ρ of $\{0, \dots, p-1\}$ such that $D_k^\alpha \sim \Delta_{\rho k}$ for $0 \le k < p$. Since $\mathrm{Int}(D_k^\alpha, \Delta_{\rho k}) = \mathrm{Int}(F^\alpha, \gamma F)$, we may set $N_{D_k} := M_F$. To determine δD_k, note that

$$M_F D_0^\alpha(g_i) M_F^{-1} = \Delta_\ell(g_i) = \chi^\ell(g_i G_{i-1}) \Delta_0(g_i)$$

for a unique integer ℓ with $0 \le \ell < p$, where χ is a nontrivial representation of G_i/G_{i-1}. To compute ℓ, we just need to compare a nonzero entry of both sides of the above e-monomial matrix equation. We then set $\delta D_0 := \Delta_\ell$. For other values of k we can determine δD_k by cyclic shifts: $D_k^\alpha = (\chi^k \otimes D_0)^\alpha = (\chi^k)^\alpha \otimes D_0^\alpha \sim (\chi^\alpha)^k \otimes (\chi^\ell \otimes \Delta_0)$. Hence $\delta D_k = \Delta_{(kq+\ell) \bmod p}$, since α is the Frobenius automorphism over $K = \mathbb{F}_q$.

Case 2. Suppose that $\pi_i F \ne F$, i.e., $F^{g_i} \not\sim F$. In Phase 1 we have already constructed $D \in \mathcal{D}$ such that $D \downarrow G_{i-1} = \bigoplus_{k=0}^{p-1} F_k$ and $F_k = \pi_i^k F$ is of degree, say, f. Since $(F \uparrow G_i)^\alpha = F^\alpha \uparrow G_i$ and $F^\alpha \sim \gamma F$, δD is the unique representation $\Delta \in \mathcal{D}$ such that γF is an irreducible constituent of $\Delta \downarrow G_{i-1}$. According to our construction, $\Delta \downarrow G_{i-1} = \bigoplus_{k=0}^{p-1} \Phi_k$ with $\Phi_k = \pi_i^k \Phi$ for some $\Phi \in \mathcal{F}$. There is a permutation ρ of $\{0, \dots, p-1\}$ such that $\gamma F_k = \Phi_{\rho k}$ as well as e-monomial matrices $M_k := M_{F_k} \in \mathrm{Int}(F_k^\alpha, \Phi_{\rho k})$. To compute $N_D \in \mathrm{Int}(D^\alpha, \delta D)$, we consider $\mathrm{Int}(D^\alpha \downarrow G_{i-1}, \delta D \downarrow G_{i-1})$. By Schur's Lemma there exist constants $d_0, \dots, d_{p-1} \in L^\times$ such that

$$N_D = (P_\rho \otimes I_f) \cdot \left(\bigoplus_{k=0}^{p-1} d_k M_k \right),$$

where P_ρ is the $p \times p$ permutation matrix whose rows have been permuted according to ρ. We may assume that $d_0 = 1$. To determine the other d_k, we use the equation

$$N_D D^\alpha(g_i) N_D^{-1} = (\delta D)(g_i). \tag{2.1}$$

According to our construction in Phase 1 there are e-monomial matrices $T_k, S_k \in L^{f \times f}$ such that

$$D(g_i) = (P_\pi \otimes I_f) \cdot \left(\bigoplus_{k=0}^{p-1} T_k \right)$$

and

$$(\delta D)(g_i) = (P_\pi \otimes I_f) \cdot \left(\bigoplus_{k=0}^{p-1} S_k \right),$$

where $\pi = (0, \ldots, p-1)$. Hence, (2.1) is equivalent to

$$(P_\pi \otimes I_f) \cdot \left(\bigoplus_{k=0}^{p-1} d_{\pi k} M_{\pi k} \right) \cdot \left(\bigoplus_{k=0}^{p-1} T_k^\alpha \right) \cdot \left(\bigoplus_{k=0}^{p-1} d_k^{-1} M_k^{-1} \right)$$
$$= (P_{\rho^{-1}\pi\rho} \otimes I_f) \cdot \left(\bigoplus_{k=0}^{p-1} S_{\rho k} \right).$$

Since $d_0 = 1$, we can successively determine $d_1, \ldots, d_{p-1}$ by comparing for each k one nonzero entry of $M_{\pi k} T_k^\alpha d_k^{-1} M_k^{-1}$ and $S_{\rho k}$.

We now compute a set $\mathcal{F}'$ of representatives of $\mathrm{Gal}(L/K)$-orbits of $\mathcal{F}$ and for each $F \in \mathcal{F}'$ with character χ_F the degree of the character field $d_F := [K(\chi_F) \colon K]$ of F as well as a matrix $S_F \in \mathrm{Int}(F^{\alpha^{d_F}}, F)$. (Note that α^{d_F} generates the Galois group of $L/K(\chi_F)$.) Notice that $\ell := d_F$ is the smallest integer m such that $F^{\alpha^m} \sim F$, i.e., the smallest m such that $\gamma^m F = F$. Furthermore, it is easily checked that

$$S_F := M_{\gamma^{\ell-1}F} M_{\gamma^{\ell-2}F}^\alpha \cdots M_F^{\alpha^{\ell-1}} \in \mathrm{Int}(F^{\alpha^\ell}, F).$$

The algorithm to compute the required data is now straightforward. We take the first representation F in $\mathcal{F}$, append it to the list $\mathcal{F}'$, and set $M := M_F$. Then we go through all $\gamma^i F$, delete them from the list $\mathcal{F}$, update $M := M_{\gamma^i F} M^\alpha$, and stop as soon as $\gamma^\ell F$ equals F, deleting F from $\mathcal{F}$ in this last step. In this way we also obtain d_F. We repeat the whole process until the list $\mathcal{F}$ is empty.

3. Realization over subfields

In this step of our algorithm we take the output of the last step and compute at first for each $F \in \mathcal{F}'$ a realization T_F of F over $K(\chi_F)$, where χ_F is the character of F. We then proceed by computing a realization of the trace of $T_F F T_F^{-1}$ over K.

It is well known that any absolutely irreducible representation of LG has a realization over its character field [3, Kapitel V, Satz 14.10]. We would like to give here a proof of this fact which builds the basis of our algorithm to find such a realization. We use the following setup: F is an irreducible representation of LG of degree f, M is the character field of F, $[L \colon M] =: \ell$, and β is a generator of $\mathrm{Gal}(L/M)$. For a matrix $A \in L^{m \times m}$, we define the norm of A by $\mathrm{N}_{L/M}(A) := A^{\beta^{\ell-1}} \cdots A$. Note that if $m \neq 1$, then the norm of A does not necessarily belong to $M^{m \times m}$.

The representations F and F^β are equivalent since they have the same character. Hence there exists an invertible matrix $S \in \mathrm{Int}(F^\beta, F)$. Suppose that there exists $T \in \mathrm{GL}(f, L)$ such that $T^{-1} T^\beta = S$. Then, $SFS^{-1} = F^\beta$ implies that TFT^{-1} is invariant under β, hence T is a realization of F over M. By Speiser's Theorem [7] mentioned in the introduction such a matrix T exists if and only if $\mathrm{N}_{L/M}(S) = I_f$. (Speiser's original proof works only over infinite fields; for a general proof, see [6, page 151].) A straightforward calculation shows that $\mathrm{N}_{L/M}(S) \in \mathrm{Int}(F, F)$. Hence, Schur's Lemma implies that $\mathrm{N}_{L/M}(S) = cI_f$ for some $c \in L$. But $\mathrm{N}_{L/M}(S)^\beta = S^{-1} \mathrm{N}_{L/M}(S) S = cI_f$, hence $c \in M$. Since L is finite, any element in M is the norm of an element in L, hence there exists $d \in L$ such that $\mathrm{N}_{L/M}(dS) = I_f$. Replacing S

by dS if necessary, we obtain the existence of T, a realization of F over its character field. (See also [3, Kapitel V, Bemerkung 14.14].)

From the second step we know $\ell := [L\colon K]/d_F$ and an e-monomial matrix $S = S_F \in \mathrm{Int}(F^\beta, F)$, $\beta = \alpha^{d_F}$. Let $S =: P_\pi \mathrm{diag}(S(1), \dots, S(f))$. We first compute some auxiliary data. Suppose that π can be written as the product of ν disjoint cycles of lengths $\ell_1, \dots, \ell_\nu$ and let $\rho_1, \dots, \rho_\nu$ be a complete set of disjoint representatives of each cycle. We compute ν, $\ell_1, \dots, \ell_\nu$ and $\rho_1, \dots, \rho_\nu$, then a nonzero entry $\gamma := \prod_{j=0}^{\ell-1} S(\pi^j 1)^{\beta^{\ell-1-j}}$ of $\mathrm{N}_{L/M}(S)$, and some element c of L satisfying $\mathrm{N}_{L/M}(c) = \gamma^{-1}$; we then replace S by cS. Now we have $\mathrm{N}_{L/M}(S) = I_f$. As ℓ_i divides the order of π and the latter divides ℓ, we have $\ell_i | \ell$. Hence, we can extract from the precomputed list Ω of primitive elements of subfields of L elements $y_1, \dots, y_\nu \in L$ such that y_i has degree ℓ_i over K. The rest of the algorithm, written in pseudo code, is now as follows (0^j means $0, \dots, 0$ j-times):

1 $t := 0$;
2 **for** $i = 1$ **to** ν **do**
3 $\quad \gamma_i := \prod_{j=0}^{\ell_i - 1} S(\pi^j \rho_i)^{\beta^{(\ell_i - 1 - j)}}$;
4 $\quad$ Compute $x_i \in L$ such that $\gamma_i = x_i^{-1} x_i^{\beta^{\ell_i}}$;
5 $\quad T[i] := (0^t, x_i, x_i y_i, \dots, x_i y_i^{\ell_i - 1}, 0^{m-t-\ell_i})^\top$;
6 $\quad$ **for** $j = 1$ **to** $\ell_i - 1$ **do**
7 $\qquad T[\pi^j \rho_i] := S(\pi^{j-1}\rho_i)^{-1} T[\pi^{j-1}\rho_i]^\beta$;
8 $\quad$ **od;**
9 $\quad t := t + \ell_i$;
10 **od**;
11 $T_F := (T[1] \mid T[2] \mid \cdots \mid T[f])$.

It is not clear in advance that the above algorithm is executable since there might be no element x_i satisfying the equation in line 4. In the following we show that such an x_i always exists and prove that the matrix T obtained by our algorithm is in fact a realization of F over M.

Let $T \in L^{f \times f}$ have columns $T[1], \dots, T[f]$. Then $TS = T^\beta$ if and only if for all $1 \le i \le \nu$ and all $1 \le j \le \ell_i$ we have

$$T[\pi^j \rho_i] = S(\pi^{j-1}\rho_i)^{-1} T[\pi^{j-1}\rho_i]^\beta. \tag{3.1}$$

This implies that $T[\rho_i] = \gamma_i^{-1} T[\pi^{\ell_i}\rho_i]^{\beta^{\ell_i}}$, hence $T[\rho_i] = \mathrm{N}_{L/M_i}(\gamma_i)^{-1} T[\rho_i]^{\beta^{\ell_i}}$ which gives $\mathrm{N}_{L/M_i}(\gamma_i) = 1$, where M_i is the fixed field of β^{ℓ_i}. Hence, by Hilbert's Theorem 90 there exists x_i satisfying the condition in line 4 and our algorithm is executable. Line 7 guarantees that (3.1) is satisfied for all $1 \le i \le \nu$, $1 \le j < \ell_i$. To see that it is also satisfied for $j = \ell_i$, we only need to check that $T[\rho_i] = \gamma_i^{-1} T[\rho_i]^{\beta^{\ell_i}}$. But this follows from the choice of x_i and the fact that y_i is fixed under β^{ℓ_i}. It remains to show that T is invertible. This is true because the Vandermonde matrix $\left((y_i^{\beta^j})^k\right)_{0 \le j,k \le \ell_i - 1}$ is invertible (since y_i has degree ℓ_i over K).

At this stage of our algorithm we have a list $\mathcal{F}'$ of representatives of $\mathrm{Gal}(L/K)$-orbits of the irreducible representations of LG, for each $F \in \mathcal{F}'$ the degree d_F of the character field of F over K, and a realization T_F of F over its character field. We know that $D_F := \bigoplus_{i=0}^{d_F-1} F^{\alpha^i}$ is equivalent to an irreducible representation of KG and that all irreducible representations of KG are obtained this way.

Let $F \in \mathcal{F}'$ be of degree f and $\tilde{F} := T_F F T_F^{-1}$. We extract from Ω a primitive element γ of the character field of F over K, i.e., an element having degree $d = d_F$ over K. Let $U := V \otimes I_f$, where V is the Vandermonde matrix $V := \left(\left(\gamma^i\right)^{\alpha^j}\right)_{0 \le i,j < d}$. It is easily verified that

$$R := U \cdot \begin{pmatrix} T_F & & & \\ & T_F^{\alpha} & & \\ & & \ddots & \\ & & & T_F^{\alpha^{d-1}} \end{pmatrix}$$

is a realization of $D = \bigoplus_{i=0}^{d-1} F^{\alpha^i}$ over K.

4. The final step and concluding remarks

Given a finite field K, a supersolvable group G of exponent e in pc-presentation, and a field extension L of K containing a primitive eth root of unity, the first two steps of our algorithm have computed a set $\mathcal{F}'$ of representatives of $\mathrm{Gal}(L/K)$-orbits of the irreducibles of LG, and for each such representation a realization of its trace over K. One possible strategy to compute the KG representations out of these data would be to represent L as the residue class ring modulo an irreducible polynomial, compute a primitive element ω of $L^\times$, replace each entry of the matrices involved by their corresponding polynomial representations, and proceed with matrix multiplication (and inversion) over L. Another strategy is to start with a representation of L as a polynomial residue class ring, and to go through all the steps of the algorithm using field arithmetic in L. Here we face the difficulty of solving equations of the type $x^d = \alpha$, where d is a divisor of $|L| - 1$. Both these strategies consume exponential time, and it seems that in practice a correct implementation of any of these strategies is rather complicated.

Nevertheless, we have implemented our algorithm in the computer algebra system GAP [5]. In this implementation the final step is performed by using a table of Jacobi logarithms for L, which needs exponential space (and time). Although it is impractical for large $|L|$, this strategy performs well for small sizes of L.

References

1. U. Baum and M. Clausen, *Computing irreducible representations of supersolvable groups*, Math. Comp. **63** (1994), 351– 359. MR **94i:**20029
2. U. Baum and M. Clausen, *Fast Fourier Transforms*, BI-Wissenschaftsverlag, Mannheim, 1993. MR **96i:**68001
3. B. Huppert, *Endliche Gruppen I*, Springer Verlag, Heidelberg, 1967. MR **37:**302
4. B. Huppert and N. Blackburn, *Finite Groups II*, Springer Verlag, New York, 1982. MR **84i:**20001a

5. M. Schönert et al., *GAP – Groups, Algorithms, and Programming*, Lehrstuhl D für Mathematik, Rheinisch Westfälische Technische Hochschule, Aachen, Germany, fourth edition, 1994.
6. J.P. Serre, *Local Fields*, Springer Verlag, New York, 1979. MR **82e:**12016
7. A. Speiser, *Zahlentheoretische Sätze aus der Gruppentheorie*, Math. Zeit. **5** (1919), 1–6.

Institut für Informatik, Römerstrasse 164, 53121 Bonn, Germany
E-mail address: `amin@cs.bonn.edu`

Institut für Informatik, Römerstrasse 164, 53121 Bonn, Germany
Current address: International Computer Science Institute, 1947 Center Street, Berkeley, California 94704–1198
E-mail address: `amin@icsi.berkeley.edu`