Finding splitting elements and maximal tori in matrix algebras

Willem A. de Graaf* Gábor Ivanyos†

August 3, 1999

Abstract

Roughly speaking, a splitting element of a matrix algebra is an element with a maximal number of eigenvalues. Using randomization, splitting elements can be found very easily and have proved to be extremely useful in computational representation theory. Efficient deterministic algorithms for finding splitting elements have been known only for special cases. In this paper we present a deterministic polynomial time method for constructing splitting elements and closely related subalgebras – maximal tori – which works over a wide range of ground fields.

1 Introduction

Let K be a field and let $\overline{K}$ denote the algebraic closure of K. A splitting element of a matrix algebra $A \leq M_n(K)$ is an element $a \in A$ with a maximal number of eigenvalues among the elements of $\overline{K} \otimes_K A \leq M_n(\overline{K})$. Being objects in "sufficiently general position", splitting elements have proved to be very useful in computing the structure of matrix algebras. The notion (in special cases) was introduced by W. M. Eberly [E1, E2, E3]. He observed that if K is sufficiently large then a random element of A has a good chance to being a splitting element. Below we give a partial list of applications.

[E2] Assume that K is an algebraic number field or a sufficiently large finite field and A is a commutative and semisimple algebra over K. Then a splitting element $a \in A$ is nothing else than a generator of A as a K-algebra; therefore it can be used to construct an isomorphism $A \cong K[x]/(f(x))$ where $f(x)$ is the a minimal polynomial of a. An immediate consequence is a randomized

*School of Mathematical and Computational Sciences, University of St Andrews, North Haugh, St Andrews, Fife, KY16 9SS, Scotland E-mail: wdg@dcs.st-and.ac.uk.

†Computer and Automation Institute, Hungarian Academy of Sciences, Lágymányosi u. 11., H-1111 Budapest, Hungary E-mail: Gabor.Ivanyos@sztaki.hu. Research supported by FKFP Grant 0612/1997, OTKA Grant 016503, and NWO-OTKA Grant N26673

polynomial time reduction from finding the simple components of algebras over K to factoring univariate polynomials over K.

[BR] Suppose that K is an algebraic number field, A is a central simple algebra over K, and a is a splitting element of A. Let α be any eigenvalue of a and $K' = K[\alpha]$. Then K' is a splitting field of A (this justifies the terminology "splitting element") and an explicit isomorphism $K' \otimes_K A \cong M_m(K')$ ($m = \sqrt{\dim_K A}$) can be constructed in polynomial time. The result is a randomized polynomial time algorithm for finding the complex irreducible representations of A.

[E3] Suppose that K is a real number field, A is a central simple algebra over K, and a is a splitting element of A. Let α be an imaginary eigenvalue of a, $\overline{\alpha}$ its complex conjugate, and $K' = K[\alpha + \overline{\alpha}, \alpha\overline{\alpha}]$. Then a minimal left ideal of $K' \otimes A$ can be constructed in polynomial time. The consequence is a randomized polynomial time algorithm for computing the local indices of simple algebras over number fields at the infinite primes. Complemented with the method of [IR] for the local indices at the finite primes, this gives an algorithm for calculating the global index.

[I] Recently, splitting elements appeared to be useful in computing the radical of algebras (in particular over fields of prime characteristic.)

The only point where randomization is used in the algorithms listed above is finding a splitting element. Hence these methods can be derandomized as soon as we have an efficient deterministic algorithm for exhibiting splitting elements. The first result in this direction is due to Rónyai [Ró]. Using the theory of orders, he showed how to find a splitting element in a simple algebra over a number field deterministically. In [GIR] methods for the closely related task of constructing Cartan subalgebras of certain Lie algebras are presented. Unfortunately, the cases efficiently treated in [GIR] do not include Lie algebras of associative algebras over certain important infinite fields of positive characteristic (e.g., global function fields).

In this paper we present a method which works over arbitrary sufficiently large ground fields, including infinite fields of positive characteristic. The algorithm performs a polynomial number of arithmetic steps. Over ground fields admitting efficient linear algebra (e.g., finite fields, number fields, function fields of bounded transcendence degree) a modified version of the algorithm runs in time which is polynomial in the size of the input.

The paper is structured as follows. The remainder of this section is devoted to a brief description of the computational models we work with. In Section 2 we summarize some known facts about splitting elements and subalgebras generated by semisimple splitting elements (the maximal tori), as well as algorithms for some subtasks. A general method for finding a maximal torus is described in Section 3. The algorithm is based on Lemma 2 which provides us with a test for maximality of a torus. In Section 4 we show how to modify the algorithm in order to keep the sizes of the intermediate objects small. This results in a polynomial time algorithm over ground fields admitting efficient linear algebra. We have implemented the algorithm in Version 4 of the GAP programming system [GAP97]. A comparison of the running time of the present algorithm to two other methods (the randomized algorithm and an algorithm using Cartan subalgebras, based on the methods of [GIR]) is given in Section 5.

We assume an encoding for the elements of the ground field K. The matrix algebra $A \leq M_n(K)$ is assumed to be given by an array of n by n matrices over K which form a K-linear basis of A. One usual way to measure the cost of an algorithm is the *uniform cost* which is the number of arithmetic operations (including zero tests) performed in the worst case, measured in the total number of field elements representing the input. The basic subtasks of linear algebra (performing the usual matrix operations, computing characteristic and minimal polynomials, solving systems of linear equations) used as subroutines of our algorithms have solutions of polynomial uniform cost (c.f. [BP]).

A more realistic measure is the *Boolean cost* where the total number of "bit operations" is taken into account. The Boolean cost of an algorithm is compared to the total number of bits representing the input. Working in this model it is natural to require that the costs of the field operations of K are bounded by a polynomial of the total size of the operands. By a *polynomial time* algorithm we mean an algorithm the Boolean cost of which is bounded by a polynomial in the size of the input. The Boolean cost of an algorithm of polynomial uniform cost is also polynomial provided that the size of the intermediate objects computed by the algorithm remains below a bound which is a polynomial in the input size.

We require that the principal tasks of linear algebra can be accomplished in polynomial time. Also, in order to exclude pathological encodings it is natural to assume that for every integer k, $\min\{|K|, k\}$ distinct elements of K can be listed in time $k^{O(1)}$ (in the Boolean sense). We refer to fields satisfying these requirements as fields admitting *efficient linear algebra*. Finite fields, algebraic number fields, or function fields over these which have transcendence degree bounded by a constant (using an appropriate, "standard" encoding for the elements of K) admit efficient linear algebra [F, IRSz]. We note that the standard encoding for a finite field K represents elements with $\theta(\log |K|)$ bits and hence an algorithm of polynomial uniform cost has automatically polynomial Boolean cost as well.

2 Preliminaries

We fix some notation used throughout the paper. K is always a field, $\overline{K}$ denotes its algebraic closure and $A \leq M_n(K)$ is a matrix algebra over K containing the identity matrix. For a matrix $a \in M_n(K)$ we denote by $d(a)$ the number of eigenvalues of a (in $\overline{K}$). Obviously, $d(a)$ is the degree of the squarefree part of the characteristic (or minimal) polynomial of a over $\overline{K}$. By $K[a]$ we denote the matrix algebra over K generated by a and the identity matrix. Then $\dim_K K[a]$ is the degree of the minimal polynomial of a. Let

$$a^* = \begin{cases} a_s & \text{if } \operatorname{char} K = 0 \text{ or } \operatorname{char} K > n, \\ a^{p^{\lceil \log_p n \rceil}} & \text{if } 0 < \operatorname{char} K = p < n. \end{cases}$$

Here a_s stands for the semisimple part from the Jordan decomposition of a. It is immediate that $d(a) = d(a^*) = \dim_K K[a^*]$. If $0 < \operatorname{char} K < n$ then a^* can be computed in a obvious way. In the remaining cases first we compute the squarefree part of the characteristic polynomial of a and use the Newton-Hensel iteration for computing a_s. The detailed description as well as an analysis of the algorithm is given in [BBCIL] or [Be]. It is shown that the method requires $n^{O(1)}$ arithmetical operations, and the Boolean cost is also a polynomial of the input size provided that systems of linear equations over K can be solved in polynomial time.

By a *torus* over K or a K-torus we mean a finite dimensional commutative algebra with identity which is separable over K. Recall that a finite dimensional commutative K-algebra T is separable over K if T is a direct sum of separable extension fields of K (c.f. [P], Section 10.7). A commutative matrix algebra $T \leq M_n(K)$ containing the identity matrix is a torus if and only if every element of T is a semisimple matrix. Also, the matrices in T can be simultaneously diagonalized over $\overline{K}$: there exists a matrix $u \in GL_n(\overline{K})$ such that $u^{-1}au$ is a diagonal matrix for every $a \in T$. Obviously $d(a) \leq \dim_K T$ and equality holds if and only if $T = K[a]$. If K is sufficiently large then equality can be reached: the idea of the usual proof of Steinitz's theorem about simple extensions of fields (c.f. [Ba], Theorem 3.6.1.) is applicable to this situation as well. It means that $a \in T$ is a splitting element of T if and only if a is a generator of T.

The proof mentioned above suggests the following simple method for finding a splitting element of T. Assume that $b_1, \ldots, b_s$ form a system of algebra generators of T and Ω is a subset of K such that $|\Omega| > (\dim_K T)(\dim_K T - 1)$. Assume that we have an element $a \in T$ such that $d(a) < \dim_K T$. Then we can choose $b \in \{b_1, \ldots, b_s\} \setminus K[a]$ and replace a with an element c of the form $c = a + \beta b$ $(\beta \in \Omega)$ such that $d(c) > d(a)$. In fact, the set $\{a + \beta b | \beta \in \Omega\}$ contains at least one generator of the subalgebra $K[a, b]$. In the worst case the algorithm requires calculation of $d(c)$ for $O((\dim_K T)^3)$ elements $c \in \{\alpha_1 b_1 + \ldots + \alpha_s b_s | \alpha_1, \ldots, \alpha_s \in \Omega\}$.

We remark that if K is a finite field such that $\dim_K T \leq |K|$ but $|K| \leq (\dim_K T)(\dim_K T - 1)$ then by [FR] the Wedderburn decomposition of T can be computed in time $(\dim_K T)^{O(1)}$ and the decomposition can be used to find a splitting element of T efficiently.

Now let A be a finite dimensional (associative) algebra over K with identity element 1_A. A K-torus $T \leq A$ in the algebra A is called a maximal K-torus of A if there exists no torus $T' \leq A$ such that $T < T'$. Below we summarize some known facts about maximal tori. By the Wedderburn-Malcev principal theorem (c.f. [P], Section 11.6), T is a maximal torus if and only if $T + Rad(A)$ is a maximal torus of $A/Rad(A)$. It follows that we can calculate the dimension of a maximal torus of A once we know the dimension of maximal tori of the simple components of $A/Rad(A)$.

Assume that A is a simple algebra over K with center Z and let K' be the separable closure of K in Z. It is obvious that $K' = T \cap Z$ for every maximal torus T in A, and a commutative subalgebra T of A containing K' is a torus over K if and only if T is a torus over K'. Let $T \leq A$ be a torus containing K'. Then TZ is a commutative semisimple subalgebra and hence $\dim_{K'} T = \dim_Z ZT \leq \sqrt{\dim_Z A}$. On the other hand, there exists a Z-torus U of A such that $\dim_Z(U) = \sqrt{\dim_Z A}$ (in [P], Section 13.5, this is proved for division algebras but the result can be extended to simple algebras in an obvious way). The sum of the separable closures of K' in the simple components of U is a K'-torus of U of dimension $\sqrt{\dim_Z A}$.

We obtained that the dimension of a maximal torus in an arbitrary finite dimensional K-algebra is independent of the particular choice of the torus. An easy consequence is that T is a maximal torus of A if and only if $\overline{K} \otimes_K T$ is a maximal torus of $\overline{K} \otimes_K A$.

For an arbitrary element $a \in A$ the set $T(a)$ of the elements of $K[a]$ which are separable over K form a torus. (The element b is called separable over K iff the minimal polynomial of b over K is a separable polynomial.) We remark that if $a \in M_n(K)$ is a matrix then $T(a) = K[a^*]$ and $\dim_K T(a) = d(a)$. Since generators of tori over sufficiently large fields (in particular, over $\overline{K}$) do exist we obtain the following.

Observation 1 *Let $A \leq M_n(K)$ be a matrix algebra containing the identity matrix. Then the matrix $a \in A$ is a splitting element of A if and only if $T(a)$ is a maximal torus of A.*

The discussion of this section implies that we can efficiently find a splitting element in $A \leq M_n(K)$ provided that we have an efficient algorithm for finding a maximal torus and $|K| \geq n$.

3 The main algorithm

Every torus containing a fixed torus T is obviously a subtorus of the centralizer $C_A(T)$ of T. Also, if T is not a maximal torus then there exists a semisimple matrix $a \in C_A(T) \setminus T$ and the subalgebra generated by T and a is a torus bigger than T. The following lemma provides us with an efficient way for finding such a matrix a.

Lemma 2 *Let A be a finite dimensional algebra over K, let $T \leq A$ be a torus in A, and let $S \subseteq A$ be a subset of A such that $T + TS = C_A(T)$ (e.g., S is a basis of $C_A(T)$). Then T is a maximal torus of A if and only if $T(b) \leq T$ and $T(b + c) \leq T$ for every $b, c \in S$.*

Proof. The "only if part" of the statement is obvious. To prove the reverse implication assume that $C_A(T) = T + TS$ and $T(b), T(b+c) \leq T$ for every $b, c \in S$. These conditions remain true if we extend the ground field to $\overline{K}$ and it is sufficient to prove that $\overline{K} \otimes_K T$ is a maximal $\overline{K}$-torus in $\overline{K} \otimes_K A$. Hence we can and will assume that K is algebraically closed. For every $b \in S$ we consider the Jordan decomposition $b = b_s + b_n$ of b. Since b_s generates $T(b)$ our assumption is equivalent to that b is a sum of an element of T and a nilpotent element $u(b) = b_n$. Similarly, $b+c$ is the sum of an element of T and a nilpotent element $u(b+c)$. We set $S' = \{u(b) | b \in S\}$. Then $C_A(T) = T + TS'$.

Let $\phi : C_A(T) \to M_m(K)$ be an arbitrary irreducible matrix representation of $C_A(T)$. Then $\phi(C_A(T)) = M_m(K)$ and $\phi(T) \leq Z(M_m(K)) = KI$, whence $\phi(T) = KI$ (here, I stands for the appropriate identity matrix). For every $u \in S'$, $\phi(u)$ is a nilpotent matrix and the linear span of $\phi(S') \cup \{I\}$ is $M_m(K)$. This is impossible if m is not a unit in K because in that case the identity matrix I as well as every nilpotent matrix has zero trace. Hence m is a unit in K and therefore the only scalar matrix of trace 0 is zero. For every $u, v \in \phi(S')$ the sum $u + v$ is required to be the sum of a scalar matrix and a nilpotent matrix. By comparing traces it is immediate that $u + v$ must be is nilpotent.

Assume that $m > 1$. For an arbitrary matrix $a \in M_m(K)$ we consider the characteristic polynomial $\det(a + tI) = t^m + c_1(a)t^{m-1} + c_2(a)t^{m-2} +$ terms of lower degree. Note that $c_1(a) = Tr(a)$. Amitsur's formulas [A] for the coefficients of the characteristic polynomial of a sum tell us that

$$c_2(u + v) = c_2(u) + c_2(v) + c_1(u) + c_1(v) - c_1(uv).$$

Since the matrices $u, v, u + v$ are nilpotent we have $c_2(u + v) = c_2(u) = c_2(v) = c_1(u) = c_1(v) = 0$ and hence $c_1(uv) = 0$ for every $u, v \in \phi(S')$. We obtained that $Tr(u) = Tr(uv) = 0$ for every $u, v \in \phi(S')$. By (bi-)linearity, these equalities extend to the whole linear span U of $\phi(S')$. On the other hand, from $M_m(K) = KI + U$ we infer that U must coincide with the set $sl_m(K)$ of matrices of trace zero. This is a contradiction because if $m > 1$ then $sl_m(K)$ does contain matrices u, v with $Tr(uv) \neq 0$.

What we have seen so far is that every irreducible representation ϕ of $C_A(T)$ is linear and hence $\phi(S') = (0)$. This means that $S' \subseteq Rad(C_A(T))$ therefore $TS \subseteq Rad(C_A(T))$ and $C_A(T) = T + Rad(C_A(T))$. From this we infer that T is a maximal torus of $C_A(T)$. Hence it is immediate that T is a maximal torus of A as well. $\square$

As a consequence we obtain the following.

Theorem 3 *Assume that the matrix algebra $A \leq M_n(K)$ containing the identity matrix is given by a set of matrices which form a K-basis of A. Then a basis of a maximal torus T of A can be computed by an algorithm using $n^{O(1)}$ arithmetical operations in K.*

Proof. We start with (a basis of) an initial torus T, say the trivial one generated by the identity and iterate the following step until T is proved to be maximal. We compute a basis S of the centralizer $C_A(T)$ of T by solving the system of linear equations $xa - ax = 0$ where a runs over the basis of T. For every pair $b, c \in S \cup \{0\}$ we compute $(b+c)^*$ using the method described in Section 2. Then we test whether $(b+c)^* \in T$. Again, this can be done by solving a system of linear equations. If $(b+c)^* \notin T$ for some pair b, c then we calculate a basis of the bigger torus generated by $T \cup \{b+c\}^*$ and proceed with it in place of T. Otherwise we can terminate as T is a maximal torus by the lemma. Obviously, the algorithm terminates in $O(n)$ rounds. The systems of linear equations involved are all of dimension $O(n^2)$ therefore they can be solved using $n^{O(1)}$ operations. $\square$

Unfortunately we cannot prove that the Boolean cost of this algorithm (over an infinite field K with efficient linear algebra) is polynomial. The point is that we are unable to give a polynomial bound on the sizes of the tori calculated in the course of the algorithm because of the iterative centralizer computation. However, as size problems over finite fields do not occur we immediately obtain the following.

Corollary 4 *Assume that K is a finite field. Then a maximal torus of A can be found in polynomial time. Furthermore, if $|K| \geq n$, then a splitting element of A can be computed in polynomial time.*

4 Reduction of sizes

In this section we describe a modification of the algorithm which runs in polynomial time over infinite ground fields with efficient linear algebra. The approach is analogous to the reduction technique of [GIR]. The idea is based on the fact that $d(a)$ can be described by polynomial functions on $M_n(K)$.

We observe that if $\operatorname{char} K = 0$ or $\operatorname{char} K > n$ then for an arbitrary matrix $a \in M_n(K)$ we have $d(a) = \dim_K(K[a]/Rad(K[a]))$ and

$$Rad(K[a]) = \{b \in K[a] | Tr(a^i b) = 0 \ \ (i = 0, 1, \ldots, n-1)\}$$

(c.f. [D]). From this it is immediate that $d(a)$ is the rank of the Bezoutian $Tr(a^i a^j)_{i,j=0}^{n-1}$.

Unfortunately this does not work if $0 < \operatorname{char} K = p \leq n$. In that case we have to apply the cruder approach based on $d(a) = \dim_K K[a^*]$. We obtain that $d(a) < k$ for some $1 < k \leq n$ if and only if the matrices $I = {a^*}^0, {a^*}^1, \ldots, {a^*}^{k-1}$ are linearly dependent. Recall that $a^* = a^{p^{\lceil \log_p n \rceil}}$.

Lemma 5 *Set $N = n(n-1)$ if $0 < \operatorname{char} K \leq n$, and $N = n(n-1)^3/2$ otherwise. Assume that $|K| > N$ and let Ω be a subset of $K \setminus \{0\}$ such that $|\Omega| > N$. Let $a, b \in M_n(K)$ two matrices. Then there exists an element $\omega \in \Omega$ such that $d(a + \omega b) \geq d(a)$.*

Proof. Assume that $d(a) = k$. If $\operatorname{char} K = 0$ or $\operatorname{char} K > n$ then let $\mathcal{F} \subseteq K[t]$ be the set of the determinants of the k by k minors of the matrix $Tr((a+tb)^i(a+tb)^j))_{i,j=0}^{n-1} \in M_n(K[t])$. The degrees of the polynomials in $\mathcal{F}$ are bounded by $2\sum_{j=n-k}^{n-1} j = k(2n-k-1) \leq n(n-1) = N$. We have $d(a+\omega b) < k$ if and only if ω is a common zero of $\mathcal{F}$.

If $0 < \operatorname{char} K = p \leq n$ then set $q = p^{\lceil \log_p n \rceil}$ and let $\mathcal{F} \subseteq K[t]$ consist of the determinants of the k by k minors of the matrix the rows of which are $(a+tb)^{q0}, (a+tb)^{q1} \ldots, (a+tb)^{q(k-1)}$, each considered as a row vector of length n^2. Now the degree bound is $p^l k(k-1)/2 \leq (n-1)^2 k(k-1)/2 \leq n(n-1)^3/2 = N$. Again, $d(a+\omega b) < k$ if and only ω is a common zero of $\mathcal{F}$.

In both cases, $d(a) = k$ therefore there exists at least one polynomial $f(t) \in \mathcal{F}$ such that $f(0) \neq 0$. In particular $f(t)$ is not the zero polynomial. Since $\deg f(t) \leq N$, it is impossible that $f(t)$ vanishes on the set Ω of cardinality greater then N. $\square$

We are ready to outline the modified algorithm. First we choose a set $\Omega \subseteq K$ of cardinality $N+1$ consisting of elements of "small" size. (In practice, if $\operatorname{char} K = 0$ then we can take $\Omega = \{0, \ldots, N\}$. If K is a function field over a finite field, we take $N+1$ polynomials of smallest degree.) Assume that A is given by the basis $b_1, \ldots, b_r$. Initially we can take $a = I$. We compute a basis S of the centralizer $C_A(a^*) = C_A(T(a))$ and use the test of Lemma 2. We either terminate with the conclusion that $T(a)$ is a maximal torus (or, equivalently a is a splitting element) or find an element b of the form $b = (b'+c')$ $(b', c' \in S \cup \{0\})$ such that $b^* \notin T(a)$. Then we use the method described in Section 2 to find an element $a' \in a^* + \Omega b^*$ such that $d(a') > d(a)$. We express a' in terms of the basis $b_1, \ldots, b_r$: $a' = \alpha_1 b_1 + \ldots + \alpha_r b_r$. Then, for the indices $i = 1$ in increasing order we do the following. If $\alpha_i \notin \Omega$ we replace a' with an element $a' + (\omega - \alpha_i)b_i$ $(\omega \in \Omega)$ such that $d(a' + (\omega - \alpha_i)b_i) \geq d(a)$. By Lemma 5, such an element exists and obviously can be found using $O(N) = n^{O(1)}$ trials. We achieve that the new a' is of the form $\omega_1 b_1 + \ldots + \omega_r b_r$ $(\omega_i \in \Omega)$ and $d(a') > d(a)$. We proceed with a' in place of a. The "small" coordinates guarantee that the sizes of matrices we compute with in the course of the algorithm remain below a bound which is polynomial in the size of the input. We have proved the following.

Theorem 6 *Assume that K is an infinite field with efficient linear algebra. Then a splitting element (and a maximal torus) of A can be computed in polynomial time.*

5 Evaluation

In this section we compare the method described in Section 3 to two other methods. The first of these is the randomized method mentioned briefly in the introduction. Let Ω be a sufficiently large subset of the ground field K. Let $\{a_1, \ldots, a_m\}$ be the basis of the algebra A that is input to the algorithm. Choose randomly and uniformly m elements α_i from Ω $(1 \leq i \leq m)$. Set $a = \sum_i \alpha_i a_i$. Then with large probability a^* generates a maximal torus of A. Note that this algorithm is probabilistic of Monte Carlo type: the output is correct with a probability that can be made arbitrarily close to 1.

The second method uses the Lie algebra structure of A. Let A_L denote the vector space A equipped with the multiplication $[\ ,\ ] : A_L \times A_L \to A_L$ defined by $[a, b] = ab - ba$. Then it is well known that A_L is a Lie algebra. First we calculate a Cartan subalgebra H of L (see [GIR]), then by Theorem 4.5.17 of [W], H is the centralizer of a maximal torus T of A. Then we calculate the centre $C(H)$ of H (see [BKS]). Then $T \subset C(H)$ and since $C(H)$ is a space of commuting matrices, we have that T is spanned by a^* where a runs through a basis of $C(H)$.

We have tested the methods on the algebras $M_n(\mathbf{Q})$ for various values of n. The results are displayed in Table 1.

n	dim(A)	Deterministic	Cartan	Random
8	64	8	13	3
9	81	9	26	7
10	100	16	50	14
11	121	24	93	29
12	144	31	160	58

Table 1: Running times (in seconds) for the calculation of a maximal torus of the associative algebra of all $n \times n$ matrices over $\mathbf{Q}$. "Deterministic" denotes the method described in Section 3, "Cartan" is the method using Cartan subalgebras and "Random" refers to the randomized method.

One can observe that the deterministic method performs quite well. The randomized method is very fast for small examples, but the running times of this method increase rapidly with the dimension. This is caused by the fact that a random element a will be a dense matrix (i.e., a matrix with many nonzero entries). This makes the calculation of a^* and of a basis of the torus generated by a^* very time consuming. The deterministic method, on the contrary, works mainly with sparse matrices (containing many zeros). This reduces the time needed for the linear algebra computations. Finally it is also apparent from the table that calculating a Cartan subalgebra of A_L is rather expensive compared to the other methods.

Acknowledgement

The authors are grateful to Lajos Rónyai for his useful remarks and suggestions.

References

[A] S. A. Amitsur, On the characteristic polynomial of a sum of matrices, *Linear and Multilinear Algebra 8 (1980) 177–182.*

[Ba] J. R. Bastida, Field Extensions and Galois Theory, in *Rota, G-C. (ed), Encyclopedia of Mathematics and Its Applications, Vol. 22. Cambridge University Press and Addison-Wesley, 1984.*

[BBCIL] L. Babai, R. Beals, J.-Y. Cai, G. Ivanyos, E. M. Luks, Multiplicative equations over commuting matrices, *Proc. 7th ACM-SIAM Symp. on Discrete Algorithms, (1996), 498-507.*

[Be] R. Beals, Algorithms for matrix groups and the Tits alternative, *Proc. 36th IEEE FOCS, (1995), 593-602.*

[BKS] Beck, R.E., Kolman, B. and Stewart, I.N. Computing the structure of a Lie algebra. in: Beck, R.E. and Kolman, B. eds., *Computers in Non-associative Rings and Algebras*, Academic Press, New York, 167–188 (1977).

[BP] D. Bini, V. Pan, *Polynomial and matrix computations, Vol 1 (Fundamental algorithms) (Birkhäuser, Basel, 1994).*

[BR] L. Babai, L. Rónyai, Computing irreducible representations of finite groups, *Mathematics of Computation 55, 192 (1990), 705-722.*

[D] L. E. Dickson, Algebras and their arithmetics, University of Chicago, Chicago, 1923.

[E1] W. M. Eberly, Computations for algebras and group representations, *Ph. D. Thesis, Dept. of Computer Science, University of Toronto, 1989.*

[E2] W. M. Eberly, Decomposition of algebras over finite fields and number fields, *Computational Complexity 1 (1991) 179–206.*

[E3] W. M. Eberly, Decompositions of algebras over **R** and **C**, *Computational Complexity 1, (1991), 207-230.*

[F] M. A. Frumkin; Polynomial time algorithms in the theory of linear diophantine equations, *Proc. of FCT'76, Lecture Notes in Computer Science 56, Springer-Verlag, 1976, 386-392.*

[FR] K. Friedl, L. Rónyai, Polynomial time solution of some problems in computational algebra, *Proc. 17th ACM STOC, (1985), 153-162.*

[GAP97] The GAP Group, Lehrstuhl D für Mathematik, RWTH Aachen, Germany and School of Mathematical and Computational Sciences, U. St. Andrews, Scotland. *GAP – Groups, Algorithms, and Programming, Version 4*, 1997.

[GIR] W. A. de Graaf, G. Ivanyos, L. Rónyai, Computing Cartan subalgebras of Lie algebras, *Applicable Algebra in Engineering, Communication and Computing 7, (1996), 71-90.*

[I] G. Ivanyos, Finding the radical of matrix algebras using Fitting decompositions, *Manuscript, 1998.*

[IR] G. Ivanyos, L. Rónyai, Finding maximal orders in semisimple algebras over **Q**, *Computational Complexity 3, (1993), 245-261.*

[IRSz] G. Ivanyos, L. Rónyai, Á. Szántó, Decomposition of algebras over $F_q(X_1, \ldots, X_m)$, *Applicable Algebra in Engineering, Communication and Computing 5 (1994) 71–90.*

[P] R. S. Pierce, Associative algebras *Springer-Verlag, Berlin, 1982.*

[Ró] L. Rónyai, A deterministic method for computing splitting elements in semisimple algebras over Q, *Journal of Algorithms 16, (1994), 24-32.*

for verification

[W] D. J. Winter, Abstract Lie Algebras, *M.I.T. Press, Cambridge, Mass., 1972.*