Algorithmic construction of Chevalley bases

Magaard, Kay; Wilson , Robert

DOI:
10.1112/S1461157012001180

Document Version
Publisher's PDF, also known as Version of record

Citation for published version (Harvard):
Magaard, K & Wilson , R 2012, 'Algorithmic construction of Chevalley bases', *London Mathematical Society. Journal of Computation and Mathematics*, vol. 15, pp. 436. https://doi.org/10.1112/S1461157012001180

LMS J. Comput. Math. 15 (2012) 436–443 © 2012 Authors
doi:10.1112/S1461157012001180

Algorithmic construction of Chevalley bases

K. Magaard and R. A. Wilson

Abstract

We present a new algorithm for constructing a Chevalley basis for any Chevalley Lie algebra over a finite field. This is a necessary component for some constructive recognition algorithms of exceptional quasisimple groups of Lie type. When applied to a simple Chevalley Lie algebra in characteristic $p \geqslant 5$, our algorithm has complexity involving the seventh power of the Lie rank, which is likely to be close to best possible.

1. *Introduction*

Finding a Chevalley basis for a semisimple Lie algebra over $\mathbb{C}$ amounts to diagonalizing a regular semisimple element: the eigenspaces for non-zero eigenvectors are just the 1-dimensional root spaces, and suitable eigenvectors can be chosen as described by Carter [1]. Indeed, the same is true for any Chevalley Lie algebra (that is, the Lie algebra of a split semisimple algebraic group) over any algebraically closed field. However, over a finite field the problem is much more difficult. The probability that a random regular semisimple element is split is approximately the reciprocal of the order of the Weyl group, so something better than a random search is required if we want a polynomial-time algorithm.

Let us define a *toral subalgebra* of a Lie algebra $\mathfrak{l}$ to be any abelian subalgebra consisting of semisimple elements. If $\mathfrak{t}$ is a maximal toral subalgebra which is split, then its centralizer in $\mathfrak{l}$ is a *Cartan subalgebra* $\mathfrak{c}$, and conversely, $\mathfrak{t}$ consists exactly of the semisimple elements in $\mathfrak{c}$.

Problem 1. Given a split toral subalgebra $\mathfrak{h}_0$ in a Chevalley Lie algebra $\mathfrak{l}$, find a Cartan subalgebra $\mathfrak{h}$ such that $\mathfrak{h}_0 \subset \mathfrak{h}$, and a Chevalley basis with respect to $\mathfrak{h}$.

Problem 2. Given two Cartan subalgebras $\mathfrak{h}_1, \mathfrak{h}_2$ of a semisimple Lie algebra $\mathfrak{l}$, find an element $g \in \mathrm{Aut}(\mathfrak{l})$ such that $\mathfrak{h}_1^g = \mathfrak{h}_2$.

Solutions to these problems are a necessary component for some constructive recognition algorithms of exceptional quasisimple groups of Lie type [5]. A polynomial-time Las Vegas algorithm for solving Problem 1 is given by Ryba [7], except in characteristic 2 (where indeed Problem 1 has no solution in general), and except for $\mathfrak{a}_2$ and $\mathfrak{g}_2$ in characteristic 3. This algorithm has complexity involving the eleventh power of the Lie rank of the algebra, as well as the fourth power of the logarithm of the field order, although practical implementations are apparently much faster than this suggests. He asserts that his algorithm often works in characteristic 2, but does not attempt a full analysis in that case.

Another algorithm is given by Cohen and Murray [3], with the same exceptions, with complexity (in the case when the input is an algebra corresponding to a simple algebraic group) involving the ninth power of the Lie rank. (A noteworthy feature of their algorithm is that the rate-determining step seems to be checking at each stage whether they have finished. It is possible therefore that their algorithm can be improved by a more subtle approach to this particular step.) They do not discuss the exceptional cases.

Received 23 March 2012; revised 13 June 2012.

2010 Mathematics subject classification 17B45, 20G40 (primary).

The small characteristic exceptions are discussed by Cohen and Roozemond [4], but they only consider the problem of finding a Chevalley basis once a Cartan subalgebra has been found. The problem of finding such a subalgebra in the first place is dealt with by Roozemond in [6]. Problem 2 amounts to finding a base-change matrix which maps one Chevalley basis to another, so is easily reduced to Problem 1, as will be discussed at the end of Section 2.

In this paper we propose a simpler algorithm which has better complexity than the above algorithms in the simple case. We achieve this by computing the whole Chevalley basis at once, rather than by first computing the Cartan subalgebra. We build up the Dynkin diagram one node at a time, making each connected component in full before moving on to the next. Our main theorem is as follows.

THEOREM 1. *Let $\mathfrak{l}$ be a Chevalley Lie algebra over a field of order q and characteristic $p \geqslant 5$. Suppose $\mathfrak{l}$ has Lie rank l and dimension d. Then there is a (probabilistic) algorithm to compute a Chevalley basis of $\mathfrak{l}$ in $O(ld^3 \log q)$ field operations.*

2. *The main algorithm*

Throughout, we use fraktur notation for Lie algebras and italic for the corresponding root systems and Dynkin diagrams. We assume that the characteristic of the field is at least 5. In this case our strategy is to look for a (long or short root) fundamental $\mathfrak{a}_1$, and find its Chevalley basis $\{e, f, h\}$. Since the Weyl group is transitive on roots of each length in a component, this root may be taken to be an end node of a connected component of the Dynkin diagram. Then we look for another fundamental $\mathfrak{a}_1$ which extends it to a simple rank 2 algebra (if there is one). Continuing in this way, we build up the connected component of the Dynkin diagram one node at a time. Then we iterate the procedure until all components are dealt with. Once all components are completed, we use the 'extraspecial pairs' as described by Carter [1] to complete the Chevalley basis for the corresponding simple Lie subalgebras. The algorithm in detail is as follows. (Comments on 'suitable' choices follow the algorithm.)

(1) Input: a Chevalley Lie algebra $\mathfrak{l}_0$ over a finite field of characteristic $p \geqslant 5$, and a split toral subalgebra $\mathfrak{h}_0$ (defaulting to zero).

(2) Output: a Cartan subalgebra $\mathfrak{h}$ containing $\mathfrak{h}_0$, together with the part of a Chevalley basis for $\mathfrak{l}_0$, consisting of the e_α, f_α, h_α for simple roots α, and a complete weight space decomposition $\mathcal{W}$ of $\mathfrak{l}_0$.

(3) Initialise $\mathfrak{h}_1 := 0$. Initialise $\mathfrak{h} := 0$. Initialise $\mathcal{D} := \emptyset$.

(4) If $\mathfrak{h}_0 \neq 0$, compute the weight spaces for $\mathfrak{h}_0$, and set $\mathcal{W}$ equal to the set of weight spaces, and pair the weight-spaces for opposite non-zero weights.
Else pick a random $x \in \mathfrak{l}_0$ and compute the eigenspaces of $\operatorname{ad} x$ on $\mathfrak{l}_0$, until there are some non-trivial eigenspaces with non-zero eigenvalues, and set $\mathcal{W}$ equal to this set of eigenspaces, paired as before. Adjoin to $\mathcal{W}$ the perp of $\mathcal{W}$, so that $\mathcal{W}$ spans the whole space.

(5) Repeat the following until $\mathcal{W}$ consists of a single subspace which is abelian.

(i) Using the current $\mathcal{W}$, find a fundamental (long or short root) $\mathfrak{a}_1$ subalgebra, as follows.

(a) Until there is a pair of opposite 1-dimensional members of $\mathcal{W}$, pick a pair of opposite spaces $V^+, V^- \in \mathcal{W}$ with $\dim V^+$ minimal, and pick random $y \in V^+$ and $z \in V^-$, and let $x = [y, z] \in [V^+, V^-]$, and refine the members of $\mathcal{W}$ using the eigenspaces of $\operatorname{ad} x$ and the perp. Recompute the pairing of members of $\mathcal{W}$.

(b) Pick $e \in V^+$ and $f \in V^-$, and set $h = [e, f]$. Then scale h so that $[h, e] = 2e$; then scale f so that $[e, f] = h$. Set $\mathfrak{h}_1 := \langle h \rangle$. Set $\mathcal{D} := \{e, f, h\}$.

(c) Compute the eigenspaces of $\operatorname{ad} h$. Refine $\mathcal{W}$ using these eigenspaces. Label each element of $\mathcal{W}$ by the corresponding eigenvalue of $\operatorname{ad} h$. (This label is the first coordinate of what will become the weight vector.)

(ii) Repeat until *either* roots of two different lengths have been found, *or* a maximal A_n diagram has been found.
(a) Pick a suitable label ($\mathfrak{h}_1$-weight) w where the next node of the diagram might live (if we do not already know the type of the component, pick the weight $(0, 0, \ldots, 0, -1)$).
(b) If the corresponding weight spaces are zero, then the diagram so far has type A_n for some n, and the actual type of the component can be identified from Table 1, so break to (iii).
(c) Repeat: Pick a pair V^+, V^- of non-zero opposite spaces in $\mathcal{W}$, with labels $\pm w$, and random $x \in [V^+, V^-]$, and compute eigenspaces of $\operatorname{ad} x$ on V^+ and V^-; until $\operatorname{ad} x$ has a pair of 1-dimensional eigenspaces $\langle e \rangle \subseteq V^+$, $\langle f \rangle \subseteq V^-$ for non-zero eigenvalues $\pm\lambda$.
(d) Set $h = [e, f]$. Then scale h so that $[h, e] = 2e$; then scale f so that $[e, f] = h$.
(e) Adjoin h to $\mathfrak{h}_1$. Adjoin to $\mathcal{D}$ the vectors e, f, h.
(f) Compute the eigenspaces of $\operatorname{ad} h$. Refine $\mathcal{W}$ using these eigenspaces. Append to the label of each element of $\mathcal{W}$ the corresponding eigenvalue of $\operatorname{ad} h$.
(g) Use Table 1 to identify the length of the new root. If roots of both lengths have been found, identify the type of the Dynkin diagram of the component, again from Table 1.

(iii) Until the current diagram is isomorphic to the known diagram of the component, repeat Steps 5(ii)(a, c–f), using knowledge of the embedding of the current diagram in the diagram of the full component to identify the next node to construct at each stage. In more detail:
(a) if there are roots of two lengths, extend the tail of long roots (in the case B_n), or short roots (in the case C_n), or both (in the case F_4);
(b) if the diagram is A_3, and the component is B_n or D_{n+1} for $n \geqslant 3$, extend from the middle node a tail of length $n - 2$. In the case B_n, we now have the extended Dynkin diagram, so remove the first node of the diagram;
(c) if the diagram is A_n, and the full component is D_{n+1}, extend once from the penultimate node;
(d) A_5 inside E_6: adjoin a node to the middle node;
(e) A_5 inside E_7: adjoin a tail of length 2 to the penultimate node;

TABLE 1. *Dimensions of weight spaces for h in a simple Lie algebra.*

Type	Root	$\dim V_1$	$\dim V_2$	$\dim V_3$
A_n		$2(n-1)$	1	
D_n		$4(n-2)$	1	
E_6		20	1	
E_7		32	1	
E_8		56	1	
B_n	Short	0	$2n-1$	
B_n	Long	$4n-6$	1	
C_n	Short	$4(n-2)$	3	
C_n	Long	$2(n-1)$	1	
F_4	Short	8	7	
F_4	Long	14	1	
G_2	Short	2	1	2
G_2	Long	4	1	

Note: V_λ denotes the eigenspace with eigenvalue λ. Since $\dim V_{-\lambda} = \dim V_\lambda$, we omit half the eigenvalues. We assume that the characteristic of the field is at least 7: obvious modifications of this table apply in smaller characteristics.

(f) A_7 inside E_7; adjoin a node to the middle node, and then remove the first node;
(g) A_7 inside E_8: adjoin a tail of length 2 to the penultimate node, and then remove the first node;
(h) A_8 inside E_8: adjoin a node to the ante-penultimate node, and then remove the first node;

(iv) Write out $\mathcal{D}$, and all of $\mathcal{W}$ which consists of 1-spaces labelled by non-zero weights, together with these labels.
Adjoin $\mathfrak{h}_1$ to $\mathfrak{h}$.
Remove the part of $\mathcal{W}$ which has been written out, and initialise labels to $\emptyset$. Initialise $\mathcal{D} := \emptyset$.

(v) If $\mathcal{W} = [W]$, pick a random $x \in W$ and compute the eigenspaces of $\operatorname{ad} x$ on W, until there are some non-trivial eigenspaces with non-zero eigenvalues, and set $\mathcal{W}$ equal to this set of eigenspaces, paired as before. (If this fails, then W is probably abelian, so break.) Adjoin to $\mathcal{W}$ the perp of $\mathcal{W}$, so that $\mathcal{W}$ spans the whole space.

(6) Now $\mathfrak{h}$ is a subspace of the single element of $\mathcal{W}$, so adjoin to $\mathfrak{h}$ a complement. Write out $\mathfrak{h}$.

Comments on the algorithm. In Step 5(i)(a), the $\mathfrak{a}_1$ subalgebra is a fundamental (long or short root) $\mathfrak{a}_1$, since the weight-spaces are 1-dimensional. In Step 5(ii)(a), we construct part of the diagram by repeatedly trying to attach a node to the previous one. In most cases, this means looking in the weight space corresponding to the weight $(0, \ldots, 0, -1)$. The exceptions are the cases B_n and G_2 at the first step, if a short root has been found. In these cases the appropriate weights are (-2) and (-3) respectively.

Indeed, in all cases when a short root has been found at the first step, we can find a long root at the next step by looking in V_{-2} or V_{-3}. At the termination of Step 5(ii), in the case when roots of two different lengths have been found, the dimensions of the weight spaces of a short root determine the diagram of the component, so that in Step 5(iii) we know exactly where to look for the remaining simple roots. So we may suppose that only long roots have been found. We now show that this can only happen in the cases A_n, B_n, D_n or E_n. More specifically, we prove the following.

LEMMA 2. *Let R be any connected root system, S a proper subsystem of type A_k for some k, consisting of long roots of R, and let $r_1, \ldots, r_k$ be a set of simple roots for S, labelled consecutively along the Dynkin diagram. Suppose that there is no root which is both perpendicular to $r_1, \ldots, r_{k-1}$ and not perpendicular to r_k. Then one of the following holds:*

(i) *$S = A_3$, $R = B_n$ or D_{n+1}, and $n \geqslant 3$;*
(ii) *$k \geqslant 4$ and $R = D_{k+1}$;*
(iii) *$R = E_n$ and (k, n) is one of $(5, 6)$, $(5, 7)$, $(7, 7)$, $(7, 8)$ or $(8, 8)$.*

Proof. In case C_n, we may assume the long root is $(2, 0, \ldots, 0)$, and the next root found is necessarily short. In case G_2, we may find two long roots, forming an A_2 diagram, but then we keep going and necessarily find a short root at the next step, forming an extended G_2 diagram, so the second and third roots found form a system of simple roots. In case F_4, we may find three long roots, forming an A_3 diagram, but again if we keep going we necessarily find a short root at the next step.

Next consider what can happen in the cases A_n, B_n, D_n or E_n, when we have found only long roots. If the diagram of the component is actually A_n, then by transitivity we may assume that the roots found at each stage are $(1, -1, 0, \ldots, 0)$, $(0, 1, -1, 0, \ldots, 0)$, and so on, and therefore we find the full Dynkin diagram. If the diagram of the component is B_n, we either find the same roots as in A_{n-1}, in which case the next root is necessarily short, and the full diagram has been found; or, we find three roots $(1, -1, 0, \ldots)$, $(0, 1, -1, 0, \ldots)$, $(-1, -1, 0, \ldots)$, at which

point we can go no further. Essentially the same thing happens in the case D_n, except that after we find A_{n-1} we can go no further.

We consider the cases E_6, E_7, E_8 individually. We take the root system for E_8 to consist of all vectors of shape $(\pm 1, \pm 1, 0, 0, 0, 0, 0, 0)$ (with the ± 1 coordinates in any positions) and $(\pm\frac{1}{2}, \pm\frac{1}{2}, \pm\frac{1}{2}, \pm\frac{1}{2}, \pm\frac{1}{2}, \pm\frac{1}{2}, \pm\frac{1}{2}, \pm\frac{1}{2})$ (with an even number of minus signs). Then the roots of E_7 may be taken as those whose first two coordinates are equal, and those of E_6 have the first three coordinates equal. Now in the case E_6, by transitivity we may assume the first root found is $(0, 0, 0, 1, -1, 0, 0, 0)$, whose stabilizer is S_6, that is, the Weyl group of A_5. This group is transitive on the twenty roots which have inner product -1 with the first root, so we may take the second root to be $(0, 0, 0, 0, 1, -1, 0, 0)$, which has stabilizer $S_3 \times S_3$. Again, this is transitive on the nine choices for the next root, so we take $(0, 0, 0, 0, 0, 1, -1, 0)$. At the next stage, there are four choices, falling into two orbits of size 2, represented by $(0, 0, 0, 0, 0, 0, 1, \pm 1)$. In both cases we have an A_4 diagram, and we find that there is a unique way to extend to A_5, and then a unique way to extend to E_6. (The only roots perpendicular to those already chosen are $\frac{1}{2}(1, 1, 1, 1, 1, 1, 1, 1)$ and $\frac{1}{2}(-1, -1, -1, 1, 1, 1, 1, -1)$. The former swaps $(0, 0, 0, 0, 0, 0, 1, 1)$ with $\frac{1}{2}(-1, -1, -1, -1, -1, -1, 1, 1)$ and fixes the other two, while the latter swaps $(0, 0, 0, 0, 0, 0, 1, -1)$ with $\frac{1}{2}(1, 1, 1, -1, -1, -1, 1, -1)$ and fixes the first two.) In the case $(0, 0, 0, 0, 0, 0, 1, -1)$ we must adjoin $\frac{1}{2}(1, 1, 1, -1, -1, -1, -1, 1)$ and $\frac{1}{2}(-1, -1, -1, -1, -1, -1, 1, 1)$, while in the case $(0, 0, 0, 0, 0, 0, 1, 1)$ we must adjoin $-\frac{1}{2}(1, 1, 1, 1, 1, 1, 1, 1)$ and $\frac{1}{2}(1, 1, 1, -1, -1, -1, 1, -1)$. These two cases are swapped by negating the first three coordinates and the last, which symmetry is the negative of an element of the Weyl group.

The E_7 case is similar. We start with $(0, 0, 1, -1, 0, 0, 0, 0)$, and have transitivity on the 32 roots at the next stage, so pick $(0, 0, 0, 1, -1, 0, 0, 0)$. We still have transitivity on the fifteen cases at the next stage, and on the eight cases at the one after, so we may suppose these are $(0, 0, 0, 0, 1, -1, 0, 0)$ and $(0, 0, 0, 0, 0, 1, -1, 0)$. Then we have two orbits, of sizes 1 and 3, at the next stage, and the cases $(0, 0, 0, 0, 0, 0, 1, \pm 1)$ are different. In one case we can extend two steps further, while in the other case we can go no further. Thus we have either an A_5 or an A_7 diagram.

Finally in the E_8 case we may take the first six roots to be $(1, -1, 0, 0, 0, 0, 0, 0), \ldots,$ $(0, 0, 0, 0, 0, 1, -1, 0)$, and then we have two different orbits at the next stage, represented by $(0, 0, 0, 0, 0, 0, 1, \pm 1)$. In one case we can go no further, while in the other we obtain one more node. Thus we have either A_7 or A_8. □

To summarise, at the conclusion of Step 5(ii) the cases where the diagram we have found consists only of long roots, and is not equal to the actual diagram of the component, are as follows:

Diagram found	Diagram of the component
A_3	$B_n (n \geqslant 3), D_n (n \geqslant 4)$
A_5	D_6, E_6, E_7
A_7	E_7, E_8, D_8
A_8	E_8, D_9
A_n	$D_{n+1} (n = 4, 6, n \geqslant 9)$

These cases are all dealt with in Step 5(iii).

The Chevalley basis. At completion of the main algorithm, we have obtained a Cartan subalgebra, and a complete set of root vectors for the fundamental roots and their negatives. We also have a set of vectors which are scalar multiples of the other root vectors. It remains to complete this to a Chevalley basis of the commutator subalgebra by computing the correct scalar multiples of these.

We assume that for every abstract Dynkin diagram, a choice of structure constants has been made (see [1, Chapter 4]). Then we scale each $e_{\alpha+\beta}$ in turn to ensure that $[e_\alpha, e_\beta]$ is the appropriate multiple $(0, \pm 1, \pm 2, \pm 3)$ of $e_{\alpha+\beta}$. This requires the characteristic to be at least 5 in the case of a component G_2, and at least 3 in the cases of a component B_n, C_n, F_4. In each case, to compute the scalar, it suffices to compute one non-zero coordinate of $[e_\alpha, e_\beta]$. This can be accomplished by computing just one column of $\operatorname{ad} e_\beta$ and applying it to e_α. Once all these scalars have been computed, we have a complete Chevalley basis for $[\mathfrak{l}_0, \mathfrak{l}_0]$.

Solution to Problem 2. In the case when $[\mathfrak{l}, \mathfrak{l}] = \mathfrak{l}$ we may use our algorithm with input $\mathfrak{h}_1$ to produce a Chevalley basis containing a basis of $\mathfrak{h}_1$, and again with input $\mathfrak{h}_2$. Then any linear map which takes the first basis to the second, preserving the labelling of the root system, will be an automorphism of the algebra mapping $\mathfrak{h}_1$ to $\mathfrak{h}_2$, as required.

3. *Analysis of the algorithm*

We first analyse the algorithm and its complexity in the case when the input algebra is simple and no partial Cartan subalgebra is given. Let l be the Lie rank, and $d \sim l^2$ the dimension of the algebra, and let q be the order of the field.

Computation of $\operatorname{ad} x$ for a random vector x takes $O(d^3)$ field operations. To compute a pair of eigenspaces for non-zero eigenvalues $\pm\lambda$ (which we do not compute), we use [2], which takes $O(d^3 \log q)$ field operations. Computing $[x, y]$ also takes $O(d^3)$ field operations, for example by computing $\operatorname{ad} y$ and applying it to x.

At the start of the algorithm (Step 4) we are looking for an element x such that $\operatorname{ad} x$ has a pair $\pm\lambda$ of non-zero eigenvalues. The proportion of such elements is at least a constant, say $1/3$ (see [3, Corollary 6.3]). Hence this step can be accomplished in $O(d^3 \log q)$ field operations.

In the simple case the main loop (Step 5) will be traversed only once. In Step 5(i)(a), (and similarly in Step 5(ii)(c), and the corresponding part of Step 5(iii)) the commutator $[y, z]$ is in effect a random matrix of small rank in the centralizer of the part of the Cartan subalgebra that we have seen. The statistics of this situation are at least as good as the statistics for a random element. Thus each of these steps takes a constant number of $O(d^3 \log q)$ steps. Moreover, just one of these steps is performed for each fundamental root. Hence in total this takes $O(ld^3 \log q)$ field operations.

To justify Step 5(i)(b) we need to show that e and f generate a split $\mathfrak{a}_1$ subalgebra. This follows from the Jacobi identity for x, e, f and for x, h, e. That is

$$[h, x] = [[e, f], x] = [[e, x], f] + [e, [f, x]] = [\lambda e, f] - [e, \lambda f] = 0,$$

and

$$[[h, e], x] = [[h, x], e] + [h, [e, x]] = 0 + [h, \lambda e]$$

so $[h, e]$ is a λ-eigenvector of $\operatorname{ad} x$ so is a scalar multiple of e. Hence, from the representation theory of $\mathfrak{sl}_2$ we know in particular that $\operatorname{ad} h$ is diagonalisable. Thus Step 5(i)(b) works, and takes a constant number of $O(d^3)$ steps. Moreover, the eigenvalues of $\operatorname{ad} h$ lie in $\{0, \pm 1, \pm 2, \pm 3\}$ so its eigenspaces can also be computed in $O(d^3)$ field operations. The same applies to Step 5(ii)(c–e) and the corresponding parts of Step 5(iii). Again, these are done once for each fundamental root, resulting in a total time of $O(ld^3)$.

Step 5(i)(c) consists of at most a constant number of eigenspace computations for known eigenvalues, so takes $O(d^3)$ operations. The same applies to Step 5(ii)(f–g) and the corresponding parts of Step 5(iii). So again we obtain a total time $O(ld^3)$.

Steps 5(iv) and 5(v) are book-keeping and termination so do not take significant time.

The final step of computing the scalars for each weight space for a non-simple root takes $O(d^2)$ field operations for each root. Thus this computation can be done in time $O(d^3)$.

Hence the overall complexity in the simple case is

$$O(ld^3 \log q) = O(l^7 \log q) = O(d^{3.5} \log q)$$

field operations. The proof of Theorem 1 is now complete.

4. *Non-simple algebras*

Semisimple case. We have designed our algorithm to apply to the semisimple case, by ensuring that in Step 5(i) we at least halve the dimension every time we find a new eigenspace. Hence this step needs to be applied at most $\log d$ times to find an $\mathfrak{a}_1$ in the first component. Since each application of Step 5(i) or Step 5(ii) or Step 5(iii) reduces the rank by 1, the overall complexity becomes $O(ld^3 \log d \log q)$.

Non-trivial centres. The part of the centre which is generated by commutators is part of the output of the algorithm. The rest of the centre plays no role, and we can pick an arbitrary basis for it.

Imperfect algebras. In this case, extra non-central toral elements appear in the final step of the algorithm. However, in general it is not possible to scale these to any particularly nice form. For example, such an h may act non-trivially on multiple components, and it is only possible to scale it to act canonically on one component. If the derived subalgebra has large homogeneous components and large codimension, this makes the definition of a canonical basis almost impossible.

In certain cases, however, it is possible to extend our algorithm. For example, if the derived subalgebra is simple, then there is at most one dimension of non-central torus outside the derived subalgebra, and we can make a canonical choice of element. For example, we can demand that $[e_i, h] = \delta_{i1} e_i$, where the e_i correspond to the fundamental roots.

5. *Characteristics* 2 *and* 3

Characteristic 3. The main problem in small characteristics is that in certain cases the weight spaces are not 1-dimensional. There may be additional problems for small fields. In characteristic 3 we only encounter problems with multidimensional weight spaces in the cases where the Lie algebra has a component of type $\mathfrak{g}_2$, or a simply-connected component of type $\mathfrak{a}_2$. In both these cases, there are eigenspaces of dimension 3. Consider first the case $\mathfrak{g}_2$. In this case, the short roots occur in weight spaces of dimension 1, so these are obtained with high probability in the same way as above, that is by looking for a short root $\mathfrak{a}_2$. However, the long roots are essentially invisible, since they lie in the zero weight space of the short roots. We distinguish between $\mathfrak{a}_2$ and $\mathfrak{g}_2$ by computing the image of $\operatorname{ad} e$ for one of the short roots e, and testing whether this lies in the $\mathfrak{a}_2$ algebra. If it does not, then we deduce that the algebra generated by the $\mathfrak{a}_2$ and this image is the full $\mathfrak{g}_2$, so we complete the calculation using [4].

The simply-connected $\mathfrak{a}_2$ case will only arise at the end, when we have run out of 1-dimensional eigenspaces, and only 3-dimensional eigenspaces remain. For each pair of these, we compute the algebra they generate, and find a suitable basis using [4]. See also [6]. We expect that these modifications will not affect the overall complexity of our algorithm.

The other problem which arises in characteristic 3 is that $2 = -1$, so that we cannot distinguish long roots from short roots using Table 1. In other words, Step 5(ii)(g) does not work as it stands. Instead, we determine whether the last two roots r, s obtained are of the same or different lengths by explicitly computing the commutators $[e_r, [e_r, e_s]]$ and $[e_s, [e_r, e_s]]$. If both of these are zero, the roots are the same length; otherwise, just one of them is non-zero,

and this tells us which of the two roots is short. We also cannot tell to begin with whether the first root found is short or long, and therefore we may run the algorithm to produce an A_k diagram consisting of short roots. However, if this happens, then we always run into a long root eventually. With these modifications the algorithm should run with the same complexity in characteristic 3.

Characteristic 2. We expect that a combination of our ideas with those of [4, 6] will also produce a more efficient algorithm in characteristic 2. First we briefly sketch how this might work in the simple case A_n.

(1) Take random x, until we have a 2-dimensional eigenspace of $\operatorname{ad} x$ with non-zero eigenvalue. Pick e, f at random in this eigenspace until $h = [e, f] \neq 0$.

(2) Find an eigenspace V of $\operatorname{ad} h$ with non-zero eigenvalue, and scale f and h so that the eigenvalue is 1.

(3) Let $V_e = [V, e] \cap V$ and $V_f = [V, f] \cap V$, and pick $y \in V_e$, $z \in V_f$ until $x := [y, z] \neq 0$.

(4) $\operatorname{ad} x$ acts on V_e and V_f, so intersect the eigenspaces of $\operatorname{ad} x$ with V_e and V_f. Similarly for $\operatorname{ad}(x + h)$. This gives us enough 1-dimensional spaces to define the root spaces for an $\mathfrak{a}_2$ subalgebra. Scale the vectors as far as possible.

(5) Continue in this way to generate each node of the diagram in turn.

More generally, there is no pairing of weight spaces, and the minimal dimension eigenspaces which we are aiming for have dimension at most 8 (see [4, Table 1]). If we modify Step 5(i)(a) by taking $V^+ = V^- \in \mathcal{W}$ then we will reach such a small-dimensional eigenspace in at most $\log d$ steps. If this dimension is not 2 or 4 then the component is of bounded rank, and the methods of [4] suffice. In the other cases, we can analyse the subalgebra generated by this eigenspace in the same way as in [4], or as suggested above in the dimension 2 case. We then extend to the whole component by a modified version of Step 5(ii): we know which eigenspace $V = V^+ = V^-$ to look in, and if this has dimension 2 we proceed as suggested in Step 4 of the A_n algorithm above. In the dimension 4 case we again split the eigenspace according to the actions of the unipotent elements already found.

However, in general in characteristic 2, not every split toral subalgebra is contained in a split maximal toral subalgebra, and therefore a heuristic algorithm such as we suggest may fail to produce a Cartan subalgebra. It may produce a maximal split toral subalgebra which is contained only in a non-split maximal toral subalgebra.

References

1. R. W. Carter, *Simple groups of Lie type* (Wiley, 1972).

2. F. Celler and C. R. Leedham-Green, 'Calculating the order of an invertible matrix', *Groups and computation II* (American Mathematical Society, providence, RI, 1997) 55–60.

3. A. M. Cohen and S. H. Murray, 'An algorithm for Lang's theorem', *J. Algebra* 322 (2009) 675–702.

4. A. M. Cohen and D. Roozemond, 'Computing Chevalley bases in small characteristics', *J. Algebra* 322 (2009) 703–721.

5. W. M. Kantor and K. Magaard, 'Black box exceptional groups of Lie type', *Trans. Amer. Math. Soc.* (to appear).

6. D. Roozemond, 'Computing maximal split toral subalgebras of Lie algebras over fields of small characteristic', *J. Symbolic Comput.* 50 (2013) 335–349.

7. A. J. E. Ryba, 'Computer construction of split Cartan subalgebras', *J. Algebra* 309 (2007) 455–483.

K. Magaard
School of Mathematics
University of Birmingham
Birmingham B15 2TT
United Kingdom

K.Magaard@bham.ac.uk

R. A. Wilson
School of Mathematical Sciences
Queen Mary, University of London
Mile End Road, London E1 4NS
United Kingdom

R.A.Wilson@qmul.ac.uk