COMPUTATIONS FOR ALGEBRAS AND GROUP REPRESENTATIONS

by

Wayne Eberly

A Thesis submitted in conformity with the requirements
for the Degree of Doctor of Philosophy in the
University of Toronto

Abstract.

This thesis is an examination of several problems which arise in the structure theory of associative algebras and matrix representations of groups, from the standpoint of computational complexity.

We begin with computational problems corresponding to the Wedderburn decomposition of matrix algebras over a field — the computation of a basis for the radical, of bases for the simple components of a semi-simple algebra, and the expression of a simple algebra as a full matrix ring over a division algebra. Building on the work of Friedl and Rónyai, we present a simple and efficient probabilistic algorithm for the computation of simple components of a semi-simple algebra. We also present probabilistic algorithms for the decomposition of simple algebras over $\mathbb{C}$ and $\mathbb{R}$. If the inputs for these problems are algebraic numbers, and are represented as elements of a single number field, then the decompositions can be performed using exact (symbolic, rather than numerical) computations, in polynomial time and with small probability of failure. We also note that these computational problems *cannot* be solved reliably by strictly numerical methods, when numerical approximations of arbitrary real or complex numbers are allowed as input: the corresponding decision problems are undecidable.

It is well known that the problems of decomposing matrix algebras and of decomposing matrix representations of finitely generated groups are closely related. We state formal reductions between these computational problems in order to obtain efficient algorithms for the problem of deciding whether a matrix representation is completely reducible, for the computation of isotypic components of a completely reducible matrix representation (Serre's "canonical decomposition"), and for computation of a set of irreducible components of a completely reducible representation over $\mathbb{C}$ and over $\mathbb{R}$ (Serre's "explicit decomposition"). Again, we obtain efficient probabilistic algorithms for exact solutions of problems, where previous algorithms have computed numerical estimates.

We continue by considering the computation of character tables (and single entries of these tables) for various classes of groups. We provide analysis of (slightly modified) algorithms which have been observed to perform well in practice. In particular, we analyse Burnside's method, to conclude that a character table over $\mathbb{C}$ can be computed from a Cayley table for G, using time polynomial in the number n of elements of G. Examining a generalisation of Dixon's method, we show that a character table for G over $\mathbb{C}$ can also be computed using space polylogarithmic in n and time polynomial in n — or, in parallel, using time polylogarithmic in n, with a polynomial number of processors. We analyse a combinatorial method for the computation of an entry of the character table of the symmetric group S_n over $\mathbb{C}$ to show that this problem is in PSPACE. We also note that, in the worst case, this algorithm requires time in $\Omega(2^{2\lfloor\sqrt{n}\rfloor}/\lfloor\sqrt{n}\rfloor)$.

Acknowledgements

I would like to thank my thesis committee of Allan Borodin, Steve Cook, Faith Fich, Eugene Luks, and David Rowe for their valuable suggestions and criticism.

My thesis supervisor, Joachim von zur Gathen, suggested the topic and has been of great help throughout the preparation of the thesis. I hope to be able to pose such insightful questions and comments myself someday.

I had a number of helpful discussions with Mark Giesbrecht, Andrew Granville, Jim Hoover, Bill Kantor, Rudi Mathon, Jim McInnes, and Lajos Rónyai, whom I now thank.

The Natural Sciences and Engineering Research Council of Canada, the University of Toronto, and my family all provided financial support while this thesis was being written.

Finally, for keeping me aware of the world outside my office, I thank Paul Beame, Josh and Laurie Benaloh, Richard Cleve, Hazel Everett, Thornton Gianfredi, Lloyd Grinham, Arvind and Lorene Gupta, Jim Hoover, Bruce Kapron, Valerie King, Paul Meyer, Naomi Nishimura, Stephen Pope, Jane Power, Prabhakar Ragde, Murray Sherk, Dan Simon, Gord Turpin, Dave Wilkes, and the friendly staff of Don Valley North Toyota.

Table of Contents

Introduction.

This thesis is an examination of several problems in the structure theory of associative algebras and matrix representations of groups from the standpoint of computational complexity. The algebraic theory suggests numerous computational problems; we find applications for these problems in areas including crystallography and atomic and nuclear spectroscopy. No provably correct and efficient algorithms are known for some of these problems. In other cases, we have algorithms which have been observed to perform well in practice, but whose worst case (or average case) performance have not been analysed — at least, not by the standards of complexity theory. Our goals, then, include the discovery of new, provably efficient algorithms for these problems, or the analysis and (where possible) improvement of existing algorithms. We also look for reductions between problems, allowing us, in some cases, to conclude that problems are intractable — that no efficient algorithms for them exist.

We examine algorithms which perform exact computations — which accept "symbolic" representations of inputs such as integers, rational numbers, or (more generally) algebraic numbers, and which return symbolic representations of outputs. Such algorithms are of use if we have available "symbolic" representations of our inputs — and this is the case for problems such as the analysis of the regular matrix representation, or the computation of the character table over $\mathbb{C}$, of a finite group (such as the symmetry group for a physical system), and for a number of problems concerning the computation of characters of linear and other continuous groups. The symbolic outputs returned by these algorithms can be used to obtain numerical estimates of the real or complex numbers being represented of arbitrarily high accuracy. Unlike fixed precision numerical estimates, these symbolic representations can also be used reliably to determine the sign of small real numbers, and to decide whether such a number is nonzero.

We should note that our condition that exact representations of inputs be available (so that these symbolic methods are applicable) will be unrealistic in many cases. Clearly, we should also look for algorithms which accept numerical estimates of inputs (preferably, with a bound on the error in these estimates also stated), and which either return accurate estimates of the outputs (again, with a bound on error provided) or indicate that such estimates cannot be determined from the inputs, as is the case, for example, when we attempt to solve a system of linear equations which is numerically singular. Such algorithms would be applicable to problems for which exact representations of inputs are not easily obtained; we suspect that these algorithms would also be more efficient than corresponding symbolic methods. A variety of numerical algorithms for the problems we discuss have been proposed; we leave the job of proving that these algorithms are correct and efficient (as described above), or of finding better numerical algorithms, for further work.

The first problem we face, when considering symbolic computations for these algebraic problems, is that of choosing a model of computation in which we can consider exact computations for problems (and, in particular, which allows us to solve decision problems involving "zero tests" correctly) while considering computations for matrix algebras and representations of groups over $\mathbb{R}$ and $\mathbb{C}$ — those problems which arise most often in physical applications. As we note in Section 1, strictly numerical computations are not sufficient for the correct solution of some intuitively simple problems. For example, we cannot decide reliably whether the polynomial $x^2 + \epsilon$ is squarefree or irreducible in $\mathbb{R}[x]$, given only a numerical estimate for a (small) real number ϵ. On the other hand, we cannot represent an arbitrary real or complex number uniquely using a finite string of 0's and 1's. We deal with this problem by assuming that our inputs are algebraic numbers (whose minimal polynomials over $\mathbb{Q}$ are known). We note again that this is a nontrivial restriction: it implies that transcendental numbers such as π and e cannot be included in the input for the problems considered. However, we can still consider (and solve) the problems associated with finite and continuous groups mentioned above. Again, we *cannot* guarantee reliable solutions for the associated problems using arbitrary real or complex numbers as inputs, by strictly numerical methods.

The model of computation is discussed in detail in Section 1. We also note that several problems with efficient solutions over number fields (in particular, the solution of systems of linear equations, and the factorisation of univariate polynomials) can also be solved efficiently under this model. The results require very minor modifications of proofs which have appeared elsewhere; we include them here because we make repeated use of these operations (in our nonstandard model) later in the thesis.

Section 2 of the thesis is concerned with the decomposition of matrix algebras — specifically, the computation of a basis for the radical of a matrix algebra; the computation of bases for the simple components of a semi-simple matrix algebra; and the decomposition of a simple algebra (that is, the expression of the algebra as a full matrix ring over a division algebra). We consider both Boolean computations over "concrete" fields, such as finite fields, number fields, and (for the model described in Section 1) over $\mathbb{C}$ and $\mathbb{R}$, and arithmetic (or "algebraic") computations over a more general class of fields (namely, over perfect fields).

We build on the work of Friedl and Rónyai [43], and of Rónyai [102]–[104]. After reviewing their algorithms for the computation of the radical, and for the computation of simple components of semi-simple matrix algebras over finite fields and number fields, we present an alternative (deterministic) algorithm for the computation of simple components of semi-simple algebras over number fields (and a corresponding probabilistic algorithm over finite fields), eliminating the computations over field extensions required by Friedl and Rónyai's algorithm. We also present a probabilistic algorithm for this problem, which minimises the use of factorisation (the most expensive part of the algorithm). In particular, we show that with high probability, it is sufficient to factor a single squarefree polynomial, and solve small systems of linear equations, in order to solve this problem. Since the problem of factoring squarefree polynomials is easily shown to be (NC^2) reducible to this problem, this is in some sense the best we can do. We then show how these algorithms can be applied to decompose algebras over $\mathbb{R}$ and $\mathbb{C}$.

We also consider the decomposition of simple algebras. Rónyai has presented an efficient algorithm for this problem for algebras over finite fields, as well as evidence that the problem is intractable, for decompositions of algebras over $\mathbb{Q}$ (see [103], [104]). In contrast, we present efficient probabilistic algorithms (again, using exact, rather than numerical computations, assuming inputs are algebraic) for the decomposition of simple algebras over $\mathbb{C}$ and $\mathbb{R}$. Previous iterative (numerical) algorithms for these problems may compute values whose "symbolic" representations have length exponential in the input size, when used to decompose algebras over $\mathbb{R}$ or $\mathbb{C}$.

In Section 3 we examine computations for matrix representations and characters of groups over fields. We begin with problems for which little or no information about the underlying group is available, and move on to problems for which we have more information about the structure of the group — and to problems for special classes of groups.

We begin by considering computations for matrix representations of arbitrary finitely generated groups. As in Section 2, we consider the decomposition of structures — in this case, of matrix representations, given the matrix representing each one of a set of generators for the underlying group. As in Section 2, there are three stages in the decomposition of these structures: the problem of deciding whether a matrix representation is completely reducible; of computing the "isotypic" components of a completely reducible representation (Serre's "canonical decomposition" of the representation); and of computing a set of irreducible components of an isotypic representation (Serre's "explicit decomposition"). These problems are known to be closely related to the computational problems (for matrix algebras) of Section 2 — Gabriel ([45]–[49]) decomposes matrix representations by examining related matrix algebras. While we cannot use the algorithm he presents for (polynomial-time) symbolic computations, we use his ideas to obtain formal reductions between the problems of Sections 2 and 3, in order to apply the results and algorithms of Section 2 to the problems for matrix representations of groups. In addition, these reductions can be used to conclude that Rónyai's negative results, for the decomposition of simple algebras over $\mathbb{Q}$, are applicable to the problem of computing irreducible components over $\mathbb{Q}$ of a matrix representation for a finitely generated group. (That is, assuming the extended Riemann Hypothesis, and allowing probabilistic polynomial time reductions, we can conclude that this problem is as difficult as that of factoring squarefree integers.)

Having seen that these computational problems for matrix representations are as difficult as the corresponding problems for matrix algebras, we show that a related problem — deciding whether two matrix representations over $\mathbb{Q}$ for a group G are equivalent — is provably easier than the corresponding problem for matrix algebras (again, assuming the extended Riemann hypothesis, and that factorisation of squarefree integers is difficult).

We next examine computations for matrix representations and characters of finite groups. We provide analysis for standard algorithms for the computations of character tables over $\mathbb{C}$ — Burnside's algorithm, and Dixon's modification of this method. We also present a third (new) algorithm, which we obtain by making a further modification to Dixon's algorithm. Part of this job is very easy: the standard methods generate character tables by factoring polynomials (over number fields and over finite fields, respectively), and solving systems of linear equations. We need only apply subsequent results about the complexity of these operations in order to conclude that Burnside's algorithm can be used to compute a character table using time polynomial in the size of the of the group. The analysis of Dixon's method is slightly more challenging; we apply results concerning the size of the smallest prime in an arithmetic progression in order to prove that Dixon's algorithm uses time polynomial in the input size in the worst case. Unfortunately, it appears to be necessary to apply results which assume the extended Riemann hypothesis if we are to prove that Dixon's algorithm has running time bounded by a polynomial function of the input size *with small degree*. While the new algorithm we present may conceivably be slightly *less* efficient than Dixon's original algorithm in the worst case, we note that it can be proved to be practical, in that its running time is bounded by a small degree polynomial function of the input size, as well as asymptotically efficient, without recourse to any unproved number theoretic hypotheses. We also note that Dixon's algorithm (and our new algorithm) can be implemented using a uniform family of Boolean circuits of size polynomial, and depth polylogarithmic, in the number of elements of the input group G. Thus, the problem of computing a character table over $\mathbb{C}$ from a Cayley table of a finite group is in NC.

We conclude by examining computations of characters for two special classes of groups: the symmetric groups, and the general linear groups. The additional information we have about the structure of these groups, and the nature of their representations and characters, has been used to design more efficient algorithms for the computations discussed above. However, it also allows us to pose new, much more difficult problems. The literature on computations for these problems is vast; in Section 3.4, we scratch the surface by giving a brief introduction to some of these problems and the algorithms commonly used to solve them. A straightforward analysis of one such algorithm is used to show that one of these problems is in PSPACE. Similar analyses can be used to prove that related problems are also in this complexity class. We also give a lower bound for the worst case running time for one of these algorithms: we show that a commonly used combinatorial method for the computation of entries of the character table of the symmetric group S_n computes $\Omega(2^{2\lfloor\sqrt{n}\rfloor}/\lfloor\sqrt{n}\rfloor)$ intermediate values in order to generate a specific entry in this table; in contrast, the input for this problem has size $O(n \log n)$.

We should acknowledge a number of sources. The texts of Curtis and Reiner [31], Serre [112], Jacobson [67], [68], and van der Waerden [117], [118] present the mathematical foundations on which our computational problems are based — and occasionally provide constructive proofs which are easily converted to algorithms for these problems. Friedl and Rónyai [43], and Rónyai [102]–[104] have previously considered some of the problems we discuss from the point of view of computational complexity; we make use of their techniques and build on their algorithms. Much of the literature on computational group theory deals with these problems, presenting algorithms which have been observed to perform well in practice, and, less frequently, with formal proofs of correctness and efficiency. In particular, we have made use of the work of Dixon [34], [35] and Gabriel [45]–[49] when considering computations for matrix representations and character tables of groups. The surveys of Butler [16], Cannon [17], and Neubüser [93] include discussions of algorithms for the computations of character tables; the bibliography of Felsch [40] of papers in computational group theory has also been useful. Finally, we should note that discussions of the physical applications of group theory often discuss the computations of characters, and the decomposition of matrix representations, for finite and continuous groups. We have made use of the very readable discussions of the representation theory of the symmetric groups and of the linear groups, of Hamermesh [61] and Wybourne [119]. Leech and Newman [79] also discuss the physical applications of the algebraic problems we consider.

1. Symbolic Computations

In this section, we provide those details about the model of computation and the representation and manipulation of elements of various fields which are to be discussed in later sections. The model of computation is discussed in Section 1.1. In Section 1.2, we describe the representation of elements of various domains to be used for Boolean computations. We also discuss "arithmetic" representations of elements of field extensions. Sections 1.3–1.6 deal with computations which arise frequently in the rest of the thesis: the solution of a system of linear equations over a field, and the factorisation of a univariate polynomial over a field.

Most of this material is standard and is included for the sake of completeness. There are a few minor exceptions: We note in Section 1.4 that Landau's algorithm for the factorisation of polynomials over number fields can be adapted to produce an "arithmetic" reduction from the factorisation of polynomials whose coefficients lie in an algebraic extension of a perfect field F to factorisation of polynomials in $F[x]$, and that a similar reduction can be obtained for the squarefree decomposition of polynomials (see [76] for Landau's algorithm). In Section 1.5, we show that algorithms for the isolation of complex roots of integer polynomials can be applied to isolate the complex roots of a polynomial whose coefficients lie in an arbitrary number field. These extensions of results are routine (and are of little interest on their own). The method of representing algebraic numbers within $\mathbb{R}$ or $\mathbb{C}$ (introduced in Section 1.2 and used in Section 1.6) may be of more interest. It is based on the method used by Collins and his collaborators for quantifier elimination over real closed fields (in [24], [4]), but permits the representation of nonreal algebraic numbers. We show in later sections that it is more useful for the problems we discuss than numerical methods for representing members of $\mathbb{R}$ or $\mathbb{C}$.

1.1. Models of Computation

We begin by specifying the (standard) models of computation and measures of costs which we have in mind when describing algorithms for algebraic problems. When specifying algorithms in the rest of the thesis, we will generally use a high level ("Pascal-like") language, leaving out the details of implementing these algorithms in the models described below.

Many of the problems we examine use elements of some (arbitrary) field F as input, and produce elements of this field as output. The *arithmetic complexity* of an algorithm for such a problem is independent of both the representation of field elements and the implementation of field arithmetic. Field operations ($+$, $-$, $\times$, and $\div$), "zero tests" (taking a single element $\alpha \in F$ as input and producing the Boolean value *true* if $\alpha = 0$, and *false* otherwise), and "selections" (between two field elements on the basis of a third, Boolean, input) are all considered to have unit cost. Hence, we count the number of these steps which are performed in order to measure the (sequential) time used by an algorithm. When measuring (sequential) space, we count the number of field elements stored. Time and space are measured as functions of the number of (field) inputs for the algorithm.

We also consider the sequential *Boolean complexity* of algorithms for problems over specific fields (such as finite fields, $\mathbb{Q}$, algebraic number fields, $\mathbb{R}$, and $\mathbb{C}$). We assume that field elements are processed as strings of Boolean values ("bits"). We count the number of Boolean operations ($\wedge$, $\vee$, and $\neg$) performed when measuring (sequential) time, and count the number of bits which must be stored when measuring (sequential) space. Time and space are measured as functions of the *length* of the input — that is, the number of bits used to represent the input. The Boolean complexity clearly depends on the method of representation of field elements, and the implementation of field arithmetic — and hence on the field over which we are computing. It provides a more realistic measure of complexity for algorithms over specific fields.

We use a *random access machine* (RAM) as our model of complexity when considering the sequential complexity of algorithms and problems. Random access machines (for integer and Boolean computations) are discussed in detail by Aho, Hopcroft, and Ullman [3]. For the reader unfamiliar with this concept, imagine an abstraction of a general purpose computer, with an unbounded random access memory and a finite set of data and address registers. When discussing Boolean computations, we assume that each data register and each memory location can contain a single bit, each address register can contain a memory address, and that the instruction set includes instructions for input and output of bits, storage and retrieval of values between the registers and memory, and the Boolean operations discussed above. When discussing arithmetic complexity over some field F, we add a second set of registers and memory locations which can contain elements of the field F, as well as instructions for input, output, and storage of field elements and for the arithmetic operations described above.

We are interested in the *parallel complexity* of algorithms and problems as well. That is, we wish to consider the cost of implementing an algorithm when a large number of processors can work together. It is customary to consider the *parallel time* and the *number of processors* used when measuring parallel complexity (rather than time and space, as for sequential computations). Again, we take an "algebraic" or "structured" approach, assuming that a single processor can perform any field operation in a single time step, in order to measure the *parallel arithmetic complexity* of an algorithm; or we can take an "unstructured" approach by measuring the *parallel Boolean complexity* of a computation over a specific field, assuming that each processor can perform a Boolean operation in constant time. As in the sequential case, the (parallel) Boolean complexity will provide a more realistic measure of complexity for algorithms over specific fields.

We use *families of circuits* as our model of parallel computation. An algorithm is represented by a family of circuits, with one circuit for each possible input size. Each circuit is an acyclic directed graph, with operations labeling nodes, and with edges between nodes representing flow of data. The *depth* of the circuit — the length of the longest directed path in the circuit — is a measure of the parallel time required by the algorithm, while the *width* of the circuit measures the number of processors used. We use families of *arithmetic-Boolean circuits* as our model of parallel arithmetic computation; these are discussed in more detail by von zur Gathen [54]. We use families of *Boolean circuits* for our model of parallel Boolean computation; these are discussed by Cook [28].

One difficulty with the circuit model of computation is that it is a nonuniform model. Different circuits are required for different input sizes; if we make no restrictions on the type of circuits allowed, then we can find families of circuits which solve unreasonably hard (in some cases, undecidable) problems. We overcome this problem by considering *uniformity conditions* — restrictions on the resources which can be used to construct the n^{th} circuit of a circuit family. Unless stated otherwise, families of circuits discussed in this thesis are log-space uniform, or *L-uniform:* a description of the circuit for input size n can be constructed using space $O(\log n)$. We will also consider families of circuits which are polynomial-time uniform, or *P-uniform:* families for which a description of the circuit for input size n can be constructed using time $n^{O(1)}$. Uniformity criteria are discussed in more detail by von zur Gathen [54] (for families of arithmetic-Boolean circuits), and by Cook [28], and Ruzzo [106] (for families of Boolean circuits).

In some cases we will not obtain an efficient algorithm for a problem P_1. Instead, we will show that it could be solved efficiently, if we had an efficient algorithm for a second problem, P_2. Formally, we exhibit a *reduction* from P_1 to P_2, by producing an algorithm for P_1 which requires the solution of one or more instances of P_2, and which is efficient — assuming that these instances of P_2 can be solved quickly. We use families of *oracle circuits* — families of arithmetic-Boolean or Boolean circuits which include *oracle nodes* solving instances of the problem P_2 — as a model for these reductions. Again, these are discussed in more detail by von zur Gathen [54] and by Cook [28].

We should also note that some of the algorithms to be discussed are *probabilistic,* rather than *deterministic.* A positive "error tolerance", ϵ, is included as part of the input. Probabilistic Boolean algorithms use a source of random bits (charging unit cost for each bit used), while probabilistic arithmetic algorithms make random choices from a finite subset of the ground field (whose size may depend on the error tolerance). Probabilistic algorithms associate to any input a *probability distribution* of possible outputs; a probabilistic algorithm is considered to be correct if it computes a valid output (rather than returning an invalid one, or reporting *failure*) with probability at least $1 - \epsilon$. (In fact, the probabilistic algorithms to be discussed here will report failure, rather than returning an invalid answer.) Clearly, efficient deterministic algorithms are preferable to probabilistic algorithms; however, we will consider some problem for which the only efficient algorithms known are probabilistic.

In general, we consider "exact", or "symbolic" solutions of algebraic problems, rather than "numeric" solutions. We assume that inputs specify unique field values (unlike floating point approximations of real numbers). For example, the rational number $\frac{1}{3}$ is represented by the ordered pair of integers (1, 3), rather than an approximation such as 0.3333333. Arithmetic is exact. This approach has several advantages: We can avoid numerical considerations such as the possibility of overflow or underflow of values, or the effect of rounding error on the accuracy of calculations. Because arithmetic is exact, the task of proving correctness of our algorithms is simplified. When we perform computations over $\mathbb{R}$ or $\mathbb{C}$, we produce "symbolic" output which can be used to generate decimal approximations of arbitrarily high accuracy. The disadvantage of this approach is the high overhead required for exact computations. We discuss representations and algorithms for exact arithmetic over fields in Section 1.2.

1.2. Symbolic Computations Over Fields

We consider the "exact" representation of field elements, and the cost of arithmetic, for several fields. With the exception of our representation of (algebraic) complex numbers, all the representations discussed here are standard.

We first consider Boolean computations.

(i) $F = \mathbb{Q}$. The rational number $\frac{a}{b}$ ($a, b \in \mathbb{Z}, b > 0$) is represented by the ordered pair (a, b). For sequential computation, it is usually assumed that integers a and b be relatively prime, since the computation of the greatest common divisor of a numerator and denominator, and division of each by this divisor, can be performed efficiently. We do not make this restriction when considering parallel computations, because no efficient parallel algorithm for the computation of this greatest common divisor is known.

Sequential algorithms for arithmetic ($+$, $-$, $\times$, $\div$) over $\mathbb{Q}$ are discussed by Collins, Mignotte, and Winkler [27]. Aho, Hopcroft, and Ullman [3] discuss the time required for integer arithmetic. They state results which imply that addition, multiplication, and division of rational numbers having representations of length N can be performed using $O(N \log^2 N \log\log N)$ Boolean operations; see also Knuth [74]. If we drop the requirement that the numerator a and the denominator b of a rational number a/b be relatively prime, then this can be reduced to $O(N \log N \log\log N)$ Boolean operations. Savage [107] gives algorithms for addition and multiplication of N-bit integers which can be used to obtain arithmetic-Boolean circuits of size $N^{O(1)}$ and depth $O(\log N)$ for addition, multiplication, and division of rational numbers.

(ii) *Algebraic Number Fields.* These are fields $F = \mathbb{Q}[\alpha]$, where α is a root of some monic integer polynomial. Suppose $f = f_0 + f_1 x + \cdots + f_{n-1}x^{n-1} + x^n$ is the minimal polynomial of α (that is, the integer polynomial of lowest degree having α as a root); if α is the root of any monic integer polynomial, then its minimal polynomial will also be monic. The field $\mathbb{Q}[\alpha]$ is isomorphic to the field $\mathbb{Q}[x]/(f)$, and has a basis

$$1, \alpha, \alpha^2, \ldots, \alpha^{n-1}$$

over $\mathbb{Q}$. Thus an arbitrary element γ of F can be represented by a set of rational numbers $g_0, g_1, \ldots, g_{n-1}$ such that

$$\gamma = g_0 + g_1\alpha + \cdots + g_{n-1}\alpha^{n-1}.$$

Using this representation, we implement arithmetic over F by implementing arithmetic for polynomials over $\mathbb{Q}$.

The *field description* consisting of the degree and coefficients of f does not include information needed to distinguish between α and the other roots of f. While it identifies $\mathbb{Q}[\alpha]$ uniquely up to field isomorphism (since $\mathbb{Q}[\alpha] \cong \mathbb{Q}[\hat{\alpha}]$ if α

and $\hat{\alpha}$ are both roots of an irreducible polynomial $f \in \mathbb{Q}[x]$), it does not identify $\mathbb{Q}[\alpha]$ uniquely in $\mathbb{C}$. Following the approach used by Collins *et al* ([24], [4]), we make this identification unique by adding information which isolates α from its conjugates: We add representations of four complex numbers (each of the form $a + b\sqrt{-1}$ for $a, b \in \mathbb{Q}$) forming a rectangle in the complex plane with edges parallel to the real and imaginary axes, so that this rectangle encloses α, and includes no other roots of f. (We use these isolating rectangles instead of the isolating intervals in the real line used by Collins, since we allow α to be an arbitrary algebraic number.) Pinkert [96] shows that such *standard rectangles*, which isolate each of the roots of f, can be computed from the coefficients of f in polynomial time. Such a rectangle can be refined (again, using the coefficients of f, in polynomial time) to produce decimal approximations of the roots of f to arbitrarily high accuracy. Hence we can use a standard rectangle for α, and the rational numbers $g_0, g_1, \ldots, g_{n-1}$ described above, to compute decimal approximations of an element $\gamma \in \mathbb{Q}[\alpha]$ of arbitrarily high precision.

Sequential algorithms for arithmetic over algebraic number fields are discussed by Loos [87]. Parallel algorithms for arithmetic can be obtained from parallel algorithms for polynomial arithmetic over $\mathbb{Q}$. We will discuss these further when we consider arithmetic over field extensions. Pinkert's results, and other results for complex root isolation, are discussed by Collins [25].

(iii) *$F = \mathbb{R}$ and $F = \mathbb{C}$.* It is easy to show that we cannot represent "arbitrary" elements of these (uncountable) fields using finite sequences of bits. If we are to perform computations over these fields, we must either settle for the computation of numerical approximations, or restrict attention to a relatively small set of instances of our computational problem, in order to guarantee that all values computed have exact representations. We take the latter approach, considering only real and complex numbers which are algebraic over $\mathbb{Q}$ — that is, which are roots of integer polynomials. Further, we assume that inputs for algorithms are represented as elements of some number field $\mathbb{Q}[\alpha]$, as described above. This will be sufficient for the computations to be considered. In particular, we will see in Chapter 3 that every linear representation of a finite group over $\mathbb{R}$ or $\mathbb{C}$ is isomorphic to a matrix representation, where all matrix entries are elements of such a number field.

For algorithms using only field arithmetic ($+$, $-$, $\times$, and $\div$), it will follow that intermediate values and outputs will also lie in the number field $\mathbb{Q}[\alpha]$. However, several of the algorithms we present include factorisation of polynomials over $\mathbb{R}$ and $\mathbb{C}$. This operation generally produces real (or complex) values lying outside the number field $\mathbb{Q}[\alpha]$. The outputs will be algebraic — they will belong to some larger number field $\mathbb{Q}[\beta]$. Unfortunately, the degree of the extension $\mathbb{Q}[\beta]$ over $\mathbb{Q}$ (and the size of a "field description" for this number field) will generally be exponential in the size of the input. We produce a representation which identifies the outputs of our algorithms exactly, which can be computed efficiently, and which can still be used to produce arbitrarily close decimal approximations of the outputs, by representing these values as elements of an extension $\mathbb{Q}[\alpha_1, \alpha_2, \ldots, \alpha_k]$, where each of the generators $\alpha_1, \alpha_2, \ldots, \alpha_k$ is an algebraic number represented by its minimal polynomial and an isolating rectangle in the complex plane. This scheme has an undesirable property: Algorithms for arithmetic over extensions of $\mathbb{Q}$ of this form are more complicated (and generally more expensive) than algorithms for the same computations in simple extensions of $\mathbb{Q}$. Note, for example, that the problem of deciding whether an arbitrary element of $\mathbb{Q}[\alpha_1, \alpha_2, \ldots, \alpha_k]$ is zero is nontrivial. Fortunately, the problems we consider generally decompose algebras into several components — and our algorithms embed each component in a different "simple" extension of the ground field. Thus, we can examine (and process) each of these components separately, without using arithmetic over more "general" (and complicated) algebraic extensions.

This is not the most general method of representing elements of $\mathbb{R}$ and $\mathbb{C}$. An approach which is closer to the standard use of floating point approximations is to represent a real number α (respectively, complex number) by a sequence $\alpha_1, \alpha_2, \ldots$ of elements of $\mathbb{Q}$ (respectively, of $\mathbb{Q}[\sqrt{-1}]$) such that $|\,\alpha - \alpha_n\,| < 2^{-n}$ for all $n \geq 0$. This representation by sequences is discussed in detail by Bishop and Bridges [12]. The complexity theory of this representation has been developed by a number of authors, including Ko and Friedman [75], and Hoover [63]. This representation has the advantage of admitting (some) real and complex numbers which are not algebraic over $\mathbb{Q}$ — such as π and e. Unfortunately, several basic problems (such as testing equality of real numbers) become intractable, or even undecidable, when this method of representation is used. We sacrifice some generality by considering only algebraic numbers. However, these will be sufficient for our purposes — and we will obtain efficient algorithms for problems in representation theory by doing so.

(iv) *Finite fields.* Elements of the prime field $\mathbb{F}_p$ (for prime $p > 0$) can be represented as integers between 0 and $p - 1$. Elements of the field $\mathbb{F}_{p^l}$ can be represented in several ways. In general, we use the isomorphism $\mathbb{F}_{p^l} \cong \mathbb{F}_p[x]/(f)$ (for $f \in \mathbb{F}_p[x]$ irreducible with degree l), and represent elements of F_{p^l} by polynomials of degree less than l with coefficients in F_p.

Sequential algorithms for arithmetic over finite fields are discussed by Collins, Mignotte, and Winkler [27]. Applying the results for integer arithmetic discussed in Aho, Hopcroft, and Ullman [3], we see that for the above representation, we can perform addition in a finite prime field $\mathbb{F}_p$ using $O(N)$ Boolean operations, for input size N; multiplication can be performed using $O(N \log N \log\log N)$ Boolean operations; and division over this finite field can be performed using $O(N \log^2 N \log\log N)$ Boolean operations. No efficient (polylogarithmic depth)

> parallel algorithm is known for inversion of elements of $\mathbb{F}_p$ using this representation of field elements; we obtain efficient parallel algorithms by using a more general representation of elements of $\mathbb{F}_p$ by numerator-denominator pairs. Now basic arithmetic over $\mathbb{F}_p$ is similar to arithmetic over $\mathbb{Q}$. We need only find an efficient algorithm for integer division with remainder (by the characteristic p) to obtain efficient parallel algorithms for arithmetic. Reif [100] presents such an algorithm; this can be used to obtain L-uniform families of Boolean circuits of size $N^{O(1)}$ and depth $O(\log N \log\log N)$ (for input size N) for addition, multiplication, and division over $\mathbb{F}_p$. An algorithm for integer division with remainder presented by Beame, Cook, and Hoover [7] can be used to obtain P-uniform families of Boolean circuits of size $N^{O(1)}$ and depth $O(\log N)$ for these problems. Parallel algorithms for arithmetic over $\mathbb{F}_{p^l}$ (for $l > 1$) will be discussed after we consider arithmetic over field extensions (see also Eberly [38]).

We now consider the cost of arithmetic over a primitive algebraic extension of a field — that is, an extension which is generated over the ground field by a single element. We consider only the "symbolic" part of computations. The task of maintaining numerical estimates of values being computed, in order to embed these elements in $\mathbb{R}$ and $\mathbb{C}$, is considered in Sections 1.3–1.6.

Suppose now that E and F are fields, with $E \cong F[t]/(f)$, for some irreducible polynomial of degree n in $F[t]$. We represent elements of E by polynomials in $F[t]$ with degree less than n: an element γ of E is represented by the coefficients g_{n-1}, g_{n-2}, $\ldots$, g_1, $g_0 \in F$ of a polynomial

$$g = g_{n-1}t^{n-1} + g_{n-2}t^{n-2} + \cdots + g_1 t + g_0 \in F[t]$$

such that $\gamma = g(\alpha)$, where α is some (fixed) root of f in E.

We implement arithmetic over E using polynomial arithmetic in $F[t]$. We first consider arithmetic computations over the ground field F. Addition over E can be implemented using addition of polynomials in $F[t]$ with degree less than n. Clearly, n additions in F (in parallel, using an arithmetic-Boolean circuit of linear size and constant depth) are sufficient. Multiplication is slightly more complicated: if we multiply two polynomials in $F[t]$ with degree less than n, the product could have degree as large as $2(n-1)$. Hence we must divide the result by f and use the remainder as the representation of our product (in $F[t]/(f)$). To divide an element γ_1 by an element γ_2 of E, we compute the reciprocal γ_2^{-1} of γ_2 in E, then perform multiplication (by γ_1) in E. If $\gamma_2 = g(\alpha)$ for a polynomial $g \in F[t]$ with degree less than n, then $\gamma_2^{-1} = h(\alpha)$, for some polynomial $h \in F[t]$ with degree less than n such that $gh \equiv 1 \pmod{f}$ in $F[t]$. Thus the extended Euclidean algorithm can be applied (using inputs g and f) to compute the polynomial h, and to invert the element γ_2 of E.

If we use the standard (simple) algorithm for multiplication of polynomials, then we can conclude that the product of two polynomials in $F[t]$ with degree less than n can be computed using $O(n^2)$ arithmetic operations in F. It can be shown that the number of arithmetic operations required for division with remainder of polynomials of degree n is the same, to within a constant factor, as the number required for multiplication of degree n polynomials (see Chapter 8 of [3]). The number required to compute the output of the extended Euclidean algorithm given polynomials f, $g \in F[t]$ of degree n (that is, to compute $\gcd(f, g)$ and polynomials u, $v \in F[t]$ such that $uf + vg = \gcd(f, g)$) is at most $O(\log n)$ times the number of operations required for multiplication of polynomials of degree n (again, see Chapter 8 of [3]). Thus we can conclude that we can implement multiplication and division in E using $O(n^2 \log n)$ operations over F. In fact, we can do better than this: a simple recursive algorithm, using 3 multiplications of polynomials of degree $n/2$ to compute the product of two polynomials of degree n, can be used to multiply polynomials of degree n using $O(n^{\log_2 3})$ arithmetic operations in F — hence multiplications and divisions in E can be performed using $O(n^{\log_2 3} \log n)$ operations in F. If F contains an n^{th} primitive root of unity, so that a fast Fourier transform can be applied, then the cost of arithmetic over E can be reduced to $O(n \log^2 n)$ arithmetic operations in F; Schönhage [109] and Cantor and Kaltofen [18] show that this can be reduced to $O(n \log n \log \log n)$ arithmetic operations over arbitrary fields, and to arbitrary rings R for multiplication of polynomials.

We now consider parallel algorithms over F for arithmetic in E. As noted above, addition in E can be implemented using arithmetic-Boolean circuits of constant depth and linear size. The standard algorithm for multiplication of polynomials in $F[t]$ can be used to obtain arithmetic-Boolean circuits of polynomial size and depth $O(\log n)$; however, the standard algorithm for division with remainder requires polynomial size and depth $\Theta(\log^2 n)$. Combining these, we obtain arithmetic-Boolean circuits over F of polynomial size and depth $\Theta(\log^2 n)$ for multiplication in E. Reif [100] has improved the result for polynomial division with remainder, presenting arithmetic-Boolean circuits of depth $O(\log n)$ and polynomial size for this problem, assuming the field F supports a fast Fourier transform. This restriction on F can be eliminated, for computations by P-uniform families of arithmetic-Boolean circuits (see Eberly [38] for details). Borodin, von zur Gathen, and Hopcroft [14] present arithmetic-Boolean circuits of polynomial size and depth $O(\log^2 n)$ computing the output of the extended Euclidean algorithm, for polynomials in $F[t]$ of degree n. These can be used to obtain arithmetic-Boolean circuits over F of the same (asymptotic) size and depth for division in E.

We obtain efficient sequential Boolean algorithms for arithmetic over number fields and finite fields $\mathbb{F}_{p^l}$ (for $l > 1$) using the above "arithmetic" algorithms, and implementing arithmetic over the prime field ($\mathbb{Q}$ or $\mathbb{F}_p$) as discussed by Collins, Mignotte, and Winkler [27]. We also obtain reasonably efficient parallel Boolean algorithms for arithmetic over $\mathbb{Q}[t]/(f)$ and over $\mathbb{F}_{p^l}$ using a direct simulation. However, we increase the depth of our circuits by a (small) multiplicative factor — we do not obtain circuits with depth $O(\log N)$ (for input size N). We obtain L-uniform families of circuits of depth $O(\log N \log\log N)$ and size $N^{O(1)}$ or P-uniform families of circuits of depth $O(\log N)$ and size $N^{O(1)}$ for addition and multiplication, and circuits of depth $O(\log^2 N)$ and size $N^{O(1)}$ for division, by adapting efficient Boolean circuits for polynomial arithmetic over $\mathbb{Q}[t]$ or $\mathbb{F}_p[t]$. For example, given α, $\beta \in \mathbb{Q}[t]/(f)$, we compute the product $\alpha\beta$ by performing computations with elements $\hat{\alpha}$, $\hat{\beta} \in \mathbb{Q}[t]$ such that $\alpha = (\hat{\alpha} \bmod f)$ and $\beta = (\hat{\beta} \bmod f)$. We use efficient Boolean circuits for multiplication of polynomials to compute the product $\hat{\alpha}\hat{\beta}$ in $\mathbb{Q}[t]$; we then use efficient Boolean circuits for division with remainder of polynomials to compute the desired output, $\alpha\beta = (\hat{\alpha}\hat{\beta} \bmod f)$. Note again that we have efficient parallel algorithms for these problems, provided that we represent a rational number by a numerator-denominator pair of integers $(a,\, b)$ with a and b not necessarily relatively prime, and that we represent an element of $\mathbb{F}_p$ by a numerator-denominator pair $(a,\, b)$ with $a, b \in \{\, 0,\, 1,\, \ldots,\, p-1 \,\}$, $b \neq 0$. These methods are discussed in more detail by Eberly [38]. For a discussion of efficient parallel algorithms for arithmetic in $\mathbb{F}_{p^l}$ without the above "redundant" representation of elements of $\mathbb{F}_p$, when the characteristic p is small (in particular, when p is polynomial in l), see Fich and Tompa [41], Litow and Davida [85], and von zur Gathen [57].

1.3. Solving Systems of Linear Equations

Many of the algorithms to be discussed in later sections will include the solution of systems of linear equations. We now consider efficient algorithms for this problem. These return the unique solution of a nonsingular system. For a singular system, they either indicate that no solution exists, or produce a single solution together with a basis for the null space of the coefficient matrix — so that all solutions are indicated.

We discuss efficient arithmetic algorithms for this problem, as well as the use of these algorithms to obtain efficient Boolean algorithms for computations over $\mathbb{Q}$ and over finite fields. Since these "rational" problems can be solved over $\mathbb{R}$ or $\mathbb{C}$ by working within the ground field containing the inputs, the methods for computations over number fields can be applied directly to obtain solutions over these larger fields. We also consider the solution of systems of linear equations over field extensions in order to obtain arithmetic reductions from problems over field extensions to the analogous problems over the ground field.

We first consider sequential computations over a field F. Using Gaussian elimination, we can solve a system of n linear equations in n unknowns using $O(n^3)$ arithmetic operations. Surprisingly, this result is not optimal: Strassen [114] has presented a recursive algorithm for this problem which used $O(n^{\log_2 7})$ arithmetic operations. This bound has been improved repeatedly; the current best bound for this problem is $O(n^{\omega})$ arithmetic operations, for $\omega < 2.376$ (Coppersmith and Winograd [29]). Unfortunately, these asymptotically fast algorithms are not practical for reasonable input sizes: Gaussian elimination (or, perhaps, a recursive algorithm using Strassen's method for large n and Gaussian elimination for smaller values) remains the best known "practical" algorithm. Sequential arithmetic algorithms for this and for related problems are discussed in more detail in Section 4 of the survey of von zur Gathen [56].

We now consider parallel arithmetic computations. The first efficient parallel algorithm for solution of a nonsingular system of n linear equations in n unknowns over a field of characteristic zero was given by Csanky [30]. An algorithm for the solution of nonsingular systems over arbitrary fields was given by Borodin, von zur Gathen, and Hopcroft [14]. Simpler algorithms were later given by Berkowitz [9] and Chistov [22]. All of these algorithms use a polynomial number of processors and parallel time $O(\log^2 n)$. The first known parallel algorithm for computation of the rank of a matrix (over a real field) was given by Ibarra, Moran, and Rosier [65]. Borodin, von zur Gathen, and Hopcroft [14] gave an efficient probabilistic algorithm for this problem over arbitrary fields, and showed that the solution of arbitrary systems of linear equations could be reduced to this problem and to the solution of nonsingular systems. Finally, Mulmuley [91] gave an efficient deterministic algorithm for this problem which was correct for arbitrary fields, proving that a polynomial number of processors, and parallel time $O(\log^2 n)$, is sufficient for the solution of arbitrary systems of n linear equations in n unknowns.

Standard bounds on the size of the determinant of an $n \times n$ matrix can be applied to show that the values computed using the above algorithms are small (see Mignotte [89]). Thus efficient Boolean algorithms can be obtained from these arithmetic algorithms in a direct manner. It is easy to see that we can use Boolean algorithms to solve systems over prime fields using a polynomial number of processors and depth $O(\log^3 N)$ (for input size N); systems over finite algebraic extensions of prime fields can be solved using a polynomial number of processors and depth $O(\log^4 N)$. In fact, we can do better than this. Borodin, Cook, and Pippenger [13] showed that nonsingular systems of linear equations over $\mathbb{Q}$ can be solved using a polynomial number of processors and depth $O(\log^2 N)$. We obtain circuits of polynomial size and depth $O(\log^2 N)$ for solution of nonsingular systems of linear equations over $\mathbb{F}_p$ by reducing this to the problem of computing determinants of integer matrices, and applying the results of Borodin, Cook, and Pippenger [14].

The parallel algorithms discussed above require (many) more processors than the number of steps used by the best sequential algorithms for these problems. Hence it can be argued that they are impractical. Pan and Reif [94], and Galil and Pan [50], give efficient parallel algorithms for the solution of nonsingular systems of linear equations over $\mathbb{Q}$, which use slightly more time ($O(\log^3 N)$ instead of $O(\log^2 N)$), but fewer processors, than the above algorithms. Their algorithms are provably as efficient (to within a constant factor) as the best sequential algorithms.

We now consider the cost of solving systems of linear equations over primitive extensions of fields. We obtain reasonably efficient algorithms for this problem by a direct simulation of the "arithmetic" algorithms discussed above, implementing arithmetic over the field extension using operations in the ground field as discussed in Section 1.2. We obtain parallel arithmetic algorithms using less parallel time by using a slightly different reduction to computations over the ground field.

Suppose first that F is an infinite field, and $E = F[t]/(f)$, for some monic irreducible polynomial $f \in F[t]$ with degree n. Suppose also that we are given a nonsingular system of m linear equations in m unknowns over the extension E. Applying Cramer's rule, we see that we can solve this system of equations at the cost of computing determinants of $m + 1$ matrices of order m over E (in parallel), then performing a small amount of additional arithmetic in E. Suppose now that we are given an $m \times m$ matrix M with entries in $E \cong F[t]/(f)$. Since the polynomial f has degree n, we see that there is a unique $m \times m$ matrix $\hat{M}$ whose entries are polynomials in $F[t]$, each with degree less than n, such that $M = (\hat{M} \bmod f)$. Since the determinant of a matrix is a polynomial in the entries of the matrix, it follows that

$$\det M = ((\det \hat{M}) \bmod f).$$

We also note that if the entries of $\hat{M}$ have degree less than n, then det $\hat{M}$ is a polynomial in $F[t]$ with degree at most $m(n-1)$. Hence the polynomial det $\hat{M}$ is uniquely determined by its value at $m(n-1)+1$ distinct points in F. Combining these facts, we obtain the algorithm given below.

We note that polynomial interpolation in $F[t]$ and polynomial division with remainder in $F[t]$ can both be performed by solving nonsingular systems of linear equations in F, of polynomial size. Hence this algorithm gives us a reduction from computations of determinants of matrices in E (and for solution of nonsingular systems of linear equations in E) to computations of determinants in F, provided that F is sufficiently large.

Algorithm **Determinant via Evaluation-Interpolation**.

Input.
- Integers n, $m > 0$.
- The coefficients of a monic polynomial $f \in F[t]$ of degree n, which is irreducible in $F[t]$.
- The entries $a_{i\,j}$, $1 \leq i,\, j \leq n$, of a matrix $M \in M_{m\times m}(F[t]/(f))$, with entry $a_{i\,j}$ given by the coefficients $a_{i,\,j,\,n-1},\, a_{i,\,j,\,n-2},\, \ldots,\, a_{i,\,j,\,1},\, a_{i,\,j,\,0} \in F$, such that $a_{i\,j} = a_{i,\,j,\,n-1}t^{n-1} + a_{i,\,j,\,n-2}t^{n-2} + \cdots + a_{i,\,j,\,1}t + a_{i,\,j,\,0} \bmod f$.

Output.
- Values $d_{n-1},\, d_{n-2},\, \ldots,\, d_1,\, d_0 \in F$ such that det $M = d_{n-1}t^{n-1} + d_{n-2}t^{n-2} + \cdots + d_1t + d_0 \bmod f$.

(1) Perform step 2 for distinct values $\gamma_0,\, \gamma_1,\, \ldots,\, \gamma_{m(n-1)}$ in parallel.

(2) Compute the determinant λ_h of the $m \times m$ matrix $\hat{M}(\gamma_h)$, with entries $\hat{a}_{i\,j}(\gamma_h) = a_{i,\,j,\,n-1}\gamma_h^{n-1} + a_{i,\,j,\,n-2}\gamma_h^{n-2} + \cdots + a_{i,\,j,\,0} \in F$.

(3) Compute the coefficients of the (unique) polynomial $\hat{d} \in F[t]$ with degree at most $m(n-1)$ such that $\hat{d}(\gamma_h) = \lambda_h$, for λ_h as computed in step 2. Note that $\hat{d} = \det \hat{M} \in F[t]$.

(4) Use division with remainder of polynomials in $F[t]$ (dividing $\hat{d}$ by f) to compute the values $d_{n-1},\, d_{n-2},\, \ldots,\, d_1,\, d_0 \in F$ such that $d_{n-1}t^{n-1} + d_{n-2}t^{n-2} + \cdots + d_0 \equiv \hat{d} \bmod f$. Return these values.

We would like to remove the restriction that our system of linear equations in E be nonsingular. To do this, we consider computations of the rank of $m \times m$ matrices in E. Mulmuley [91] reduces this to the problem of computing determinants of matrices whose entries are polynomials over E in (new) indeterminates x and y, with degree at most 1 in x and at most $2n$ in y. If F is sufficiently large, then we can use evaluation-interpolation to reduce this to the problem of computing determinants of matrices over E. Hence we can reduce computations of the rank of matrices in E to computations of determinants of matrices in F. We apply reductions given by Borodin, von zur Gathen, and Hopcroft [14] to conclude that arbitrary $m \times m$ systems of linear equations over E can be solved at the cost of computing determinants in F — using arithmetic-Boolean circuits over F of depth $O(\log^2(mn))$ and size polynomial in mn, if the field F is sufficiently large.

If F is a small finite field, then we cannot use evaluation-interpolation as described above, because F does not include enough evaluation points. Instead, we solve the problems discussed above using Boolean computations, using Boolean circuits of polynomial size and depth $O(\log^2(nm))$. Since F is so small, we can also translate our "arithmetic" inputs into corresponding Boolean representations, perform Boolean computations to obtain Boolean representations of the desired outputs, and then translate these back to "arithmetic" values — using arithmetic-Boolean circuits of the size and depth stated above. Finally, we note that we can also solve systems of m linear equations in m unknowns over a number field $\mathbb{Q}[t]/(f)$, or over a finite field $\mathbb{F}_{p^l}$, using Boolean circuits of size $N^{O(1)}$ and depth $O(\log^2 N)$, for input size N (see Eberly [37] for details). We summarise these results in the following theorem.

Theorem 1.3.1.

(i) Let F be an arbitrary field, and let $E = F[t]/(f)$ for some irreducible polynomial $f \in F[t]$ with degree n. Systems of m linear equations in m unknowns over the extension E can be solved using arithmetic-Boolean circuits over F of size $(mn)^{O(1)}$ and depth $O(\log^2(mn))$.

(ii) If $F = \mathbb{Q}[t]/(f)$ for an irreducible polynomial $f \in \mathbb{Q}[t]$, or if $F = \mathbb{F}_{p^l}$ for some prime p and for $l > 0$, then systems of linear equations over F can be solved using Boolean circuits of depth $O(\log^2 N)$ and size $N^{O(1)}$, for input size N.

1.4. Factoring Polynomials over Number Fields and Finite Fields

We now consider the squarefree decomposition of univariate polynomials, and the factorisation of squarefree univariate polynomials, over a field F. Unlike the solution of systems of linear equations over a field, these are not purely rational problems. We do not have universal arithmetic algorithms for squarefree decomposition (over all fields) or for factorisation; instead, we have (Boolean) algorithms for these problems over several classes of fields. We review these algorithms, for computations over finite fields and number fields; computations over $\mathbb{R}$ and $\mathbb{C}$ are discussed in Sections 1.5 and 1.6. We also present reductions from the computation of squarefree decompositions of polynomials over primitive extensions of a field F, and for factorisation of squarefree polynomials over primitive extensions, to the corresponding computations for polynomials over F, for a large class of fields.

There is actually more than one "squarefree decomposition" of a polynomial in $F[x]$. We use the definition of monotone squarefree decomposition and distinct power decomposition stated by von zur Gathen [52]. Henceforth we choose $\gcd(f_1, f_2)$ to be the unique *monic* polynomial of greatest degree dividing polynomials f_1 and f_2, for $f_1, f_2 \in F[x]$, at least one nonzero.

Definition 1.4.1. Let F be a field and let $g \in F[x]$. The polynomial g is *squarefree* if there does not exist any polynomial $h \in F[x] \setminus F$ such that h^2 divides g. Let c be the leading coefficient of g, and let $h = (h_1, h_2, \ldots, h_s)$ be a sequence of monic squarefree polynomials in $F[x]$ with $h_s \neq 1$. We call h the *monotone squarefree decomposition* of g if $g = ch_1h_2 \cdots h_s$ and h_{i+1} divides h_i for $1 \leq i < s$. This decomposition is unique, and h_1 is called the *squarefree part* of g. We call h the *distinct power decomposition* of g if $g = ch_1h_2^2 \cdots h_s^s$ and $\gcd(h_i, h_j) = 1$ for $1 \leq i < j \leq s$. This decomposition is also unique.

There is an efficient sequential algorithm for the computation of squarefree decompositions of any polynomial f over a field of characteristic zero (namely, the computation of the squarefree part of f as $\gcd(f, f')$). Von zur Gathen [52] presents a parallel algorithm for computation of the squarefree decompositions of polynomials of degree n, which can be implemented using arithmetic-Boolean circuits of size $n^{O(1)}$ and depth $O(\log^2 n)$.

No such universal algorithm exists for fields of positive characteristic; the squarefree part of a polynomial f over such a field can be a *proper* divisor of $\gcd(f, f')$. Von zur Gathen [52] considers (parallel) algorithms for the squarefree decomposition of a polynomial in $\mathbb{F}_{p^l}[x]$ for any (fixed) finite field $\mathbb{F}_{p^l}$, and shows that the problem can be reduced to that of computing large powers of elements of $\mathbb{F}_{p^l}$, and of computing the greatest common divisors of polynomials in $\mathbb{F}_{p^l}$. It is clear that the methods he describes can be used to obtain an efficient sequential (Boolean) algorithm for the squarefree decomposition of polynomials over $\mathbb{F}_{p^l}$.

Efficient parallel algorithms also exist for this problem. Suppose now that $f \in \mathbb{F}_{p^l}[x]$, and that f has degree n. If p is small (in particular, if $p \leq n$), then the algorithm of Fich and Tompa [41] for exponentiation in $\mathbb{F}_{p^l}$ can be combined with the reduction given by von zur Gathen [52] to obtain an efficient parallel algorithm for the computation of the squarefree part of f. Otherwise $p > n$, the squarefree part of f is $\gcd(f,\, f')$, and the methods for fields of characteristic zero are applicable. Again, an efficient parallel algorithm for the problem can be obtained. In particular, the above methods can be applied to produce arithmetic-Boolean circuits over $\mathbb{F}_p$ of size $(nl)^{O(1)}$ and depth $O(\log^3(nl))$ for squarefree decompositions of polynomials of degree n in $\mathbb{F}_{p^l}[x]$, assuming elements of $\mathbb{F}_{p^l}$ are represented as polynomials with degree less than l and with coefficients in $\mathbb{F}_p$. No parallel algorithms have been found which use arithmetic over $\mathbb{F}_{p^l}$ (rather than $\mathbb{F}_p$), with elements of $\mathbb{F}_{p^l}$ treated atomically, and which yield circuits of depth $O(\log^k(nl))$ for a constant k. If we consider a weaker model of parallel arithmetic computation, *arithmetic circuits* (which include operations $+$, $-$, $\times$, and $\div$, but not zero tests or selections), then it can be shown that no such algorithm exists. This negative result is discussed by von zur Gathen [54], and by von zur Gathen and Seroussi [58]; it provides evidence that squarefree decomposition is a problem for which the field $\mathbb{F}_{p^l}$ *must* be considered as a field extension (at least, if we are considering efficient parallel computations).

Finally, we should note that the computation of squarefree decompositions of polynomials in a field F, using only arithmetic in F, is actually impossible for some fields. In particular, Fröhlich and Shepherdson [44] construct a field F_S from an arbitrary recursively enumerable set $S \subseteq \mathbb{N}$ such that arithmetic is effective, and such that any algorithm which can be used to decide whether quadratic polynomials in $F_S[x]$ are squarefree in $F_S[x]$ can also be used to decide membership of natural numbers in the set S. If S is recursively enumerable but not recursive, then it follows that neither of the problems of deciding membership in S, or of deciding whether quadratic polynomials are squarefree in $F_S[x]$, is decidable.

Efficient sequential algorithms have also been developed for factorisation of squarefree polynomials over a large class of fields. Berlekamp [10] presented the first polynomial time algorithm for factorisation of squarefree polynomials over finite fields vwith small characteristic. In 1970, he also presented an efficient probabilistic algorithm for factorisation over arbitrary finite fields (Berlekamp [11]). Rabin [99] and Cantor and Zassenhaus [19] have each given alternative probabilistic algorithms for this problem. The first polynomial time algorithm for factorisation of squarefree polynomials with coefficients in $\mathbb{Q}$ was presented by Lenstra, Lenstra, and Lovász [82]. This was later generalised, to produce a polynomial time algorithm for factorisation of squarefree polynomials with coefficients in algebraic number fields (Lenstra [81] and Landau [76] give two different generalisations of this result). Landau [77] gives a more comprehensive survey of results for factorisation of polynomials.

If the field F is a finite field, say $\mathbb{F}_{p^l}$, then Berlekamp's deterministic algorithm can be used to obtain arithmetic-Boolean circuits over $+\mathbb{F}_p$, or Boolean circuits, of size $(npl)^{O(1)}$ and depth $O(\log^3(npl))$ for factorisation of squarefree polynomials of degree n in $F[x]$. Von zur Gathen [52] shows that the probabilistic algorithm of Cantor and Zassenhaus [19] can be used to obtain arithmetic-Boolean circuits over $\mathbb{F}_p$ (with extra nodes producing random elements of $\mathbb{F}_p$) or Boolean circuits (with extra nodes producing random bits), of size $(nl \log p)^{O(1)}$ and depth $O(\log^2 n \log^2 l \log p)$, which successfully factor a squarefree polynomial of degree n over $\mathbb{F}_{p^l}$ with probability at least 1/2. No efficient parallel algorithms for factorisation of squarefree polynomials over $\mathbb{Q}$ or over number fields are known.

As is the case for squarefree decomposition, there exist fields F for which the factorisation of squarefree polynomials in $F[x]$ using only arithmetic in F is actually impossible. Given a set S which is recursively enumerable but not recursive, a field $\hat{F}_S$ of characteristic 3 can be constructed, with the property that any "arithmetic" algorithm deciding whether an arbitrary squarefree polynomial of degree 2 is irreducible in $\hat{F}_S[x]$ could also be used to decide membership in S.

Hence we must look for algorithms for squarefree decomposition and for factorisation of polynomials, which are correct for (and peculiar to) specific fields — or, at least, specific classes of fields. Instead of a "universal" arithmetic algorithm for factorisation over arbitrary fields, we look for relationships between the computational problems of factoring polynomials over two closely related fields. For example, Landau [76] obtains an efficient algorithm for factorisation of polynomials over number fields by reducing this to the problem of factoring polynomials with coefficients in $\mathbb{Q}$. We will show that her method generalises, and obtain a reduction from factorisation of polynomials in $E[x]$ to factorisation of polynomials in $F[x]$, where E is a primitive algebraic extension of F, for a large class of fields F. We will use this reduction in Section 2 to reduce other problems to factorisation of polynomials.

Landau's method produces a reduction which is correct for *perfect* fields, as defined below.

Definition 1.4.2. A polynomial $f \in F[x]$ is *separable* if its irreducible factors have distinct roots in an algebraic closure of F. An algebraic extension E of F is a *separable extension* of F if the minimal polynomial (over F) of every element of E is separable.

Definition 1.4.3. A field F is *perfect* if every polynomial in $F[x]$ is separable.

Any field F of characteristic zero, and any finite field F, is a perfect field. An alternative characterisation of perfect fields of positive characteristic can be used to show that the problem of deciding whether a polynomial $f \in F[x]$ is squarefree has an efficient solution for any perfect field F.

Proposition 1.4.4. A field of characteristic $p > 0$ is perfect if and only if each element of the field has a p^{th} root in the field.

For a proof of Proposition 1.4.4, see van der Waerden [117] (Section 6.9, Theorem II). It follows from this that for a perfect field F, any polynomial

$$g = \alpha_k x^{pk} + \alpha_{k-1} x^{p(k-1)} + \cdots + \alpha_1 x^p + \alpha_0 \in F[x]$$

(so that $g' = 0$) is the p^{th} power of a polynomial $h \in F[x]$.

Corollary 1.4.5. If F is perfect and $g \in F[x] \setminus F$, then g is squarefree in $F[x]$ if and only if $\gcd(g,\, g') = 1$.

We next note that the monotone squarefree decomposition and the distinct power decomposition of a polynomial $g \in E[x] = (F[t]/(f))[x]$ can be computed efficiently from g and from the squarefree part h_1 of g. These two decompositions are closely related:

Proposition 1.4.6. Let K be a field. If $(\, h_1,\, h_2,\, \ldots,\, h_s\,)$ is the monotone squarefree decomposition of a polynomial $g \in K[x]$, and $(\, k_1,\, k_2,\, \ldots,\, k_{\hat{s}}\,)$ is the distinct power decomposition of g, then $s = \hat{s}$, and if $l_i = \gcd(g,\, h_1^i)$ for $1 \le i \le s$, then $h_i = l_i / l_{i-1}$ for $1 < i \le s$, and $k_i = h_i / h_{i+1}$ for $1 \le i < s$.

In particular, Proposition 1.4.6 is correct for the case $K = E = (F[t]/(f))$.

The facts stated above are easily checked. Computation of powers of polynomials, division of polynomials, and computation of the greatest common divisor of polynomials in E can all be reduced to the solution of nonsingular systems of linear equations in E (see Borodin, von zur Gathen, and Hopcroft [14], Reif [100], and Eberly [37]). Hence we obtain the following corollary.

Corollary 1.4.7. The monotone squarefree decomposition and the distinct power decomposition of a polynomial $g \in E[x]$ of degree m can each be computed from the coefficients of g and of the squarefree part of g, using arithmetic-Boolean circuits of size polynomial in mn and of depth $O(\log^2(mn))$, for $n = [E : F]$.

Thus it is sufficient to consider computation and factorisation of the squarefree part of a polynomial $g \in E[x] = (F[t]/(f))[x]$. We perform these computations using the *norm* of the polynomial g over F, defined below.

Suppose again that $E = F[t]/(f)$, for f monic and irreducible of degree n in $F[t]$, and for F perfect. Let $\alpha_1,\, \alpha_2,\, \ldots,\, \alpha_n$ be the roots of f in an algebraic closure H of F; since f is separable, these roots are distinct. Now $E = F[t]/(f)$ is isomorphic to the field $F[\alpha_1] \subseteq H$. (In fact, if H is an algebraic closure of $E = F[t]/(f)$, then we can set $\alpha_1 = (t \bmod f)$ — so the fields E and $F[\alpha_1]$ are actually the same.) We use this isomorphism to embed E in the closure H.

Definition 1.4.8. Let $\gamma = c_0 + c_1\alpha_1 + c_2\alpha_1^2 + \cdots + c_{n-1}\alpha_1^{n-1} \in F[\alpha_1]$, for $c_0, c_1, \ldots, c_{n-1} \in F$. Since α_1 has minimal polynomial f with degree n over F, the coefficients $c_0, c_1, \ldots, c_{n-1}$ are unique. The *norm* of γ in $F[\alpha_1]$ over F, $N_{F[\alpha_1]/F}(\gamma)$, is

$$N_{F[\alpha_1]/F}(\gamma) = \prod_{i=1}^{n}(c_0 + c_1\alpha_i + c_2\alpha_i^2 + \cdots + c_{n-1}\alpha_i^{n-1}).$$

If $g = \gamma_0 + \gamma_1 x + \gamma_2 x^2 + \cdots + \gamma_m x^m \in (F[\alpha_1])[x]$, with $\gamma_j = c_{j\,0} + c_{j\,1}\alpha_1 + c_{j\,2}\alpha_1^2 + \cdots + c_{j,\,n-1}\alpha_1^{n-1} \in F[\alpha_1]$, and with $c_{j\,k} \in F$ for $0 \le j \le m$ and $0 \le k < n$, then the *norm* of the polynomial g, $N_{F[\alpha_1]/F}(g)$, is

$$N_{F[\alpha_1]/F}(g) = \prod_{i=1}^{n}\left(\sum_{j=0}^{m}\left(\sum_{k=0}^{n-1} c_{j\,k}\alpha_i^k\right)x^j\right) \in (F[\alpha_1])[x].$$

Since $N_{F[\alpha_1]/F}(\gamma)$ and $N_{F[\alpha_1]/F}(g)$ are each fixed by the Galois group of the normal closure of $F[\alpha_1]$ over F, it is clear that $N_{F[\alpha_1]/F}(\gamma) \in F$ and $N_{F[\alpha_1]/F}(g) \in F[x]$. It is also clear from the definition of $N_{F[\alpha_1]/F}(g)$ that the polynomial g divides its norm, for any $g \in (F[\alpha_1])[x]$.

Landau states a number of results about the norms of polynomials in algebraic extensions, for the case $F = \mathbb{Q}$. The results, and the proofs given by Landau, clearly generalise. We state these results as Propositions 1.4.9, 1.4.10, and 1.4.11.

Proposition 1.4.9. Let $g \in (F[\alpha_1])[x]$ be irreducible, for α_1 algebraic over F and for F perfect. Then $N_{F[\alpha_1]/F}(g)$ is a power of an irreducible polynomial in $F[x]$.

Proposition 1.4.10. Let $g \in (F[\alpha_1])[x]$, for α_1 algebraic over F and for F perfect, with leading coefficient $c \in F[\alpha_1]$, such that $N_{F[\alpha_1]/F}(g)$ is squarefree in $F[x]$. Then if $N_{F[\alpha_1]/F}(g) = h_1 h_2 \cdots h_s$ is a factorisation into irreducible polynomials in $F[x]$, then $g = c\prod_{j=1}^{s}\gcd(g,\, h_j)$ is a factorisation into irreducible polynomials in $(F[\alpha_1])[x]$.

Proposition 1.4.11. Let $g \in (F[\alpha_1])[x]$ be squarefree with degree m, for α_1 algebraic over F, F perfect, and for $[F[\alpha_1] : F] = n$. Then there are at most $(nm)^2/2$ elements s of F such that $N_{F[\alpha_1]/F}(g(x - s\alpha_1))$ is not squarefree.

These are stated by Landau [76] for the case $F = \mathbb{Q}$ as Theorems 1.4 and 1.5, and Lemma 1.6, respectively.

Since the norm is a multiplicative function, we have the following extension.

Proposition 1.4.12. *Let F be a perfect field, α_1 algebraic over F such that $[F[\alpha_1] : F] = n$, and let $g \in (F[\alpha_1])[x]$ with degree m. Suppose g has squarefree part $h \in (F[\alpha_1])[x]$ with degree k. Then there are at most $(nk)^2/2 \leq (nm)^2/2$ elements s of F such that the squarefree part of $N_{F[\alpha_1]/F}(g(x - s\alpha_1))$ in $F[x]$ has degree less than nk. For all other $s \in F$, this squarefree part has degree nk, and the squarefree part of $g(x - s\alpha_1)$ is the greatest common divisor of $g(x - s\alpha_1)$ and the squarefree part of $N_{F[\alpha_1]/F}(g(x - s\alpha_1))$.*

Proof. The norm of $h(x-s\alpha_1)$ is a divisor of the norm of $g(x-s\alpha_1)$, of degree nk. It is a consequence of Proposition 1.4.11 that $N_{F[\alpha_1]/F}(h(x-s\alpha_1))$ is squarefree for all but at most $(nk)^2/2$ elements s of F. For these "bad" choices of s, the polynomial $N_{F[\alpha_1]/F}(h(x - s\alpha_1))$ has the squarefree part of $N_{F[\alpha_1]/F}(g(x - s\alpha_1))$ as a proper divisor. For all other choices of s, it is easily checked that $N_{F[\alpha_1]/F}(h(x - s\alpha_1))$ is itself the squarefree part of $N_{F[\alpha_1]/F}(g(x - s\alpha_1))$.

Clearly, $h(x - s\alpha_1)$ is a divisor of $\gcd(N_{F[\alpha_1]/F}(h(x - s\alpha_1)),\, g(x - s\alpha_1))$. Suppose the polynomial $h(x - s\alpha_1)$ is a proper divisor. Then

$$\gcd(N_{F[\alpha_1]/F}(h(x - s\alpha_1)),\, g(x - s\alpha_1)) = l \cdot h(x - s\alpha_1),$$

for some $l \in (F[\alpha_1])[x] \setminus F[\alpha_1]$, and it is clear that l divides both of the polynomials $(N_{F[\alpha_1]/F}(h(x - s\alpha_1)))/(h(x - s\alpha_1))$ and $g(x - s\alpha_1)/h(x - s\alpha_1)$. Let $\hat{l}$ be the squarefree part of l; then $\hat{l}$ divides both $(N_{F[\alpha_1]/F}(h(x - s\alpha_1)))/(h(x - s\alpha_1))$ and the squarefree part + of $g(x - s\alpha_1)/h(x - s\alpha_1)$. Since h is the squarefree part of g, $h(x-s\alpha_1)$ is the squarefree part of $g(x-s\alpha_1)$, and it is clear that the squarefree part of $g(x-s\alpha_1)/h(x-s\alpha_1)$ divides $h(x-s\alpha_1)$. Thus $\hat{l}$ divides $h(x-s\alpha_1)$ (since it divides $g(x - s\alpha_1)/h(x - s\alpha_1)$). Since $\hat{l}$ also divides $(N_{F[\alpha_1]/F}(h(x - s\alpha_1)))/(h(x - s\alpha_1))$, $\hat{l}^2$ divides $N_{F[\alpha_1]/F}(h(x - s\alpha_1))$. Therefore, $N_{F[\alpha_1]/F}(h(x - s\alpha_1))$ is not squarefree. We conclude from this that $h(x - s\alpha_1)$ is the greatest common divisor of $g(x - s\alpha_1)$ and $N_{F[\alpha_1]/F}(h(x - s\alpha_1))$ if $N_{F[\alpha_1]/F}(h(x - s\alpha_1))$ is squarefree, as required. ■

We must show that the polynomial $N_{F[\alpha_1]/F}(g)$ can be computed efficiently if we are to use it to factor g. The method given by Landau [76] for the case $F = \mathbb{Q}$ can be used for the general case.

Definition 1.4.13. Let $h = h_r y^r + h_{r-1} y^{r-1} + \cdots + h_0 \in K[y]$, and let $k = k_s y^s + k_{s-1} y^{s-1} + \cdots + k_0 \in K[y]$, for h_i, $k_j \in K$ for $0 \le i \le r$ and $0 \le j \le s$, and for K an integral domain. The *resultant* of k and h with respect to the indeterminate y, $\mathrm{Res}_y(k, h)$, is

$$\det \begin{bmatrix} k_s & 0 & 0 & \cdots & 0 & h_r & 0 & 0 & \cdots & 0 \\ k_{s-1} & k_s & 0 & \cdots & 0 & h_{r-1} & h_r & 0 & \cdots & 0 \\ k_{s-2} & k_{s-1} & k_s & \cdots & 0 & h_{r-2} & h_{r-1} & h_r & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ k_{s-r+1} & k_{s-r+2} & & \cdots & k_s & h_{r-s+1} & h_{r-s+2} & & \cdots & h_r \\ k_{s-r} & k_{s-r+1} & & \cdots & k_{s-1} & h_{r-s} & h_{r-s+1} & & \cdots & h_{r-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & k_0 & 0 & 0 & 0 & \cdots & h_0 \end{bmatrix},$$

where the above matrix has order $r + s$, with r columns of coefficients of k, and s columns of coefficients of h.

We compute the norm of $g \in E[x] = (F[t]/(f))[x]$, for f monic and irreducible over a perfect field F, by computing the resultant of polynomials in the indeterminate t with coefficients in $K = F[x]$. Given $g \in (F[t]/(f))[x]$, let $\hat{g} \in F[t, x]$ such that $\hat{g} = \sum_{i=1}^{m} (\sum_{j=0}^{n-1} g_{i\,j} t^j) x^i$, with $g_{i\,j} \in F$ for $0 \le i \le m$ and $0 \le j < n$, and such that $g = (\hat{g} \bmod f)$.

Proposition 1.4.14. $N_{E/F}(g) = (-1)^{mn} \mathrm{Res}_t(\hat{g},\, f) \in F[x]$.

This is a direct consequence of Theorem 1 of Loos [87].

Our reductions from squarefree decomposition and factorisation of polynomials over primitive algebraic extensions of a (large) perfect field to the respective problems for polynomials over the ground field are stated on the following pages.

The two algorithms have the same general form. In each case, an irreducible polynomial $f \in F[t]$ of degree n, and a polynomial $g \in (F[t]/(f))[x]$ of degree m, are given as input. The algorithms proceed by checking sufficiently many elements s of F to ensure that the norm over F of the squarefree part of $g(x - st)$ will be squarefree for at least one of the values checked. Given such an s, the norm of $g(x - st)$ is used to obtain the squarefree decomposition of g (by the first algorithm) or the factorisation of g (by the second) in the manner indicated by Propositions 1.4.12 and 1.4.10, respectively. Correctness of the algorithms follows from these propositions, and from Proposition 1.4.11, provided that the field F has at least $1 + \lceil (nm)^2/2 \rceil$ distinct elements.

Algorithm **Squarefree Decompositions over Extensions via the Norm**

Input.
- Integers n, $m > 0$.
- The coefficients of a monic irreducible polynomial $f \in F[t]$ of degree n.
- The coefficients $c_0, c_1, \ldots, c_{m-1} \in E = F[t]/(f)$ of a monic polynomial $g = c_0 + c_1 x + \cdots + c_{m-1}x^{m-1} + x^m \in E[x]$, with coefficient c_i given by elements $c_{i,0}, c_{i,1}, \ldots, c_{i,n-1}$ of F such that $c_i = c_{i,0} + c_{i,1}t + \cdots + c_{i,n-1}t^{n-1} \bmod f$.

Output.
- Integer $k \geq 0$.
- Elements $d_{i\,j}$ of F, for $0 \leq i < k$ and $0 \leq j < n$, such that
$$h = d_0 + d_1 x + \cdots + d_{k-1}x^{k-1} + x^k \in E[x]$$
is the squarefree part of g, with
$d_i = d_{i,0} + d_{i,1}t + \cdots d_{i,n-1}t^{n-1} \bmod f$, for $0 \leq i < k$.

Let $r = 1 + \lceil (nm)^2/2 \rceil$, and let $s_1, s_2, \ldots, s_r$ be any set of distinct elements of F.

(1) Perform steps 2–3 in parallel, for $1 \leq i \leq r$.

(2) Compute the coefficients of the polynomial
$$\bar{g}_i = N_{E/F}(g(x - s_i t)) = (-1)^{mn}\mathrm{Res}_t(\hat{g}(x - s_i t),\, f) \in F[x],$$
for $\hat{g} \in F[t, x]$ with degree less than n in t such that $g = (\hat{g} \bmod f)$.

(3) Compute the degree k_i and coefficients $\bar{g}_{i,k_i-1}, \ldots, \bar{g}_{i,1}, \bar{g}_{i,0} \in F$ such that $\bar{h}_i = x^{k_i} + \bar{g}_{i,k_i-1}x^{k_i-1} + \cdots + \bar{g}_{i,1}x + \bar{g}_{i,0}$ is the squarefree part of $\bar{g}_i$ in $F[x]$.

(4) Fix j to be any integer between 1 and r such that $k_j = \max_{1 \leq i \leq r}(k_i)$.
Set $k = k_j$, and set $h = \gcd(g, \bar{h}_j(x + s_j t))$, for the polynomial $\bar{h}_j$ as computed in step 3.

(5) Return the integer k and the coefficients of the polynomial h.

Algorithm **Factorisation over Extensions via the Norm**

Input.
- Integers n, $m > 0$.
- The coefficients of a monic irreducible polynomial $f \in F[t]$ of degree n.
- The coefficients $c_0, c_1, \ldots, c_{m-1} \in E = F[t]/(f)$ of a monic squarefree polynomial $g = c_0 + c_1 x + \cdots + c_{m-1}x^{m-1} + x^m \in E[x]$ with coefficient c_i given by elements $c_{i,\,0}, c_{i,\,1}, \ldots, c_{i,\,n-1}$ of F such that $c_i = c_{i,\,0} + c_{i,\,1}t + \cdots + c_{i,\,n-1}t^{n-1} \bmod f$.

Output.
- Integers $k > 0$, and $m_1, m_2, \ldots, m_k > 0$, with $\displaystyle\sum_{h=0}^{k} m_h = m$.
- Elements $d_{h\,i\,j}$ of F, for $1 \le h \le k$, $0 \le i < m_h$, and $0 \le j < n$, such that $g = \displaystyle\prod_{h=1}^{k} d_h$ is an irreducible factorisation in $E[x]$, for polynomials $d_h = x^{m_h} + \displaystyle\sum_{i=0}^{m_h-1}\left(\sum_{j=0}^{n-1} d_{h\,i\,j}t^j\right)x^i \bmod f$.

Let $r = 1 + \lceil (nm)^2/2 \rceil$, and let $s_1, s_2, \ldots, s_r$ be any set of distinct elements of F.

(1) Perform step 2 in parallel, for $1 \le i \le r$.

(2) Compute the coefficients of the polynomial
$\bar{g}_i = N_{E/F}(g(x - s_i t)) = (-1)^{mn}\mathrm{Res}_t(\hat{g}(x - s_i t),\, f) \in F[x]$,
for $\hat{g} \in F[t, x]$ with degree less than n in t such that $g = (\hat{g} \bmod f)$.

(3) Fix l to be any integer between 1 and r such that $\gcd(\bar{g}_l, \bar{g}_l{}') = 1$ in $F[x]$.

(4) Compute a factorisation of $\bar{g}_l = \displaystyle\prod_{h=1}^{k} \bar{d}_h$ into irreducible polynomials in $F[x]$. Set k to be the number of irreducible factors of $\bar{g}_l$, and set m_h to be the degree of the factor $\bar{d}_h$ (in x), for $1 \le h \le k$.

(5) Return the integers $k, m_1, m_2, \ldots, m_h$, and the coefficients of $d_h = \gcd(g,\, \bar{d}_h(x + s_l t)) \in (F[t]/(f))[x]$, for $1 \le h \le k$.

If the field F does not have sufficiently many distinct elements for the above algorithms, so that F is $\mathbb{F}_{p^l}$ for $p^l \leq \lceil (nm)^2/2 \rceil$, then we have (different) parallel algorithms for squarefree decomposition and factorisation, which take advantage of the fact that the ground field is small. Using methods already discussed, we obtain arithmetic-Boolean circuits over F, or Boolean circuits, of depth $O(\log^3(nm))$ and of size $(nm)^{O(1)}$ for each of these problems, if F is a prime field. These algorithms are easily modified to produce arithmetic-Boolean circuits over F, or Boolean circuits, of size $(nm)^{O(1)}$ and of depth $O(\log^3(nm))$, for the general case $F = \mathbb{F}_{p^l}$, $p^l \leq \lceil (nm)^2/2 \rceil$.

We summarise these results in the following theorem.

Theorem 1.4.15. Let F be a perfect field, and let $E = F[t]/(f)$ be a primitive algebraic extension of degree n over F.

(i) The squarefree decomposition of a polynomial $g \in E[x]$ of degree m can be computed using arithmetic-Boolean circuits over F (with oracles for squarefree decomposition in $F[x]$), with depth $O(\log^3(mn))$ and size $(mn)^{O(1)}$, plus the cost of computing the squarefree parts of $(1 + \lceil (nm)^2/2 \rceil)$ polynomials of degree mn in $F[x]$, in parallel.

(ii) The irreducible factorisation of a squarefree polynomial $g \in E[x]$ of degree m can be computed using arithmetic-Boolean circuits over F (with an oracle for factorisation in $F[x]$), with depth $O(\log^3(mn))$ and size $(mn)^{O(1)}$, plus the cost of factoring a squarefree polynomial of degree mn in $F[x]$.

1.5. Isolation of Roots over Number Fields

As noted in Section 1.2, we are interested in computations over number fields viewed as subfields of $\mathbb{R}$ or $\mathbb{C}$. Our field description of a number field isomorphic to $\mathbb{Q}[t]/(f)$ (for f monic and irreducible in $\mathbb{Q}[t]$) will include a standard rectangle, isolating a single root α of f. We are performing computations over $\mathbb{Q}[\alpha]$.

When performing rational computations (such as arithmetic, or solving systems of linear equations) over $\mathbb{Q}[\alpha]$, we produce values in $\mathbb{Q}[\alpha]$, and it is sufficient to ignore the root α of f and perform computations over $\mathbb{Q}[t]/(f)$. We recover the values we want in $\mathbb{Q}[\alpha]$ by replacing t by α in the results. This is also true for squarefree decomposition and for factorisation in $(\mathbb{Q}[\alpha])[x]$. Given a polynomial $g \in (\mathbb{Q}[\alpha])[x]$, we compute its squarefree part (in $(\mathbb{Q}[\alpha])[x]$) or its irreducible factors in $(\mathbb{Q}[\alpha])[x]$, by performing computations for polynomials in $(\mathbb{Q}[t]/(f))[x]$, and replacing t by α in the coefficients of the polynomial(s) we obtain.

This is not sufficient if we want to compute isolating rectangles for the complex roots of a polynomial $g \in (\mathbb{Q}[\alpha])[x]$ of degree m: These are not generally in the ground field. In this section, we show that isolating rectangles can be obtained for these roots in polynomial time.

The problem we consider is a generalisation of one which has been well studied: the isolation of the complex roots of an integer polynomial. Pinkert's method can be used to compute isolating rectangles and numerical estimates of arbitrarily high precision for the roots of a polynomial in $\mathbb{Q}[x]$ in polynomial time. More recent methods can be used to obtain root approximations very efficiently (see Schönhage [110]). Using the methods discussed in Section 1.4, we can generate a polynomial in $\mathbb{Q}[x]$ — namely, the norm of g over $\mathbb{Q}$ — whose roots include those of $g \in (\mathbb{Q}[\alpha])[x]$. While the above methods can be used to obtain isolating rectangles for the roots of the norm of g, we are left with the problem of distinguishing between the roots of g and the remaining roots of its norm over $\mathbb{Q}$.

Suppose now that $c \in \mathbb{Q}$ is a lower bound for the separation of distinct roots of the norm of g over $\mathbb{Q}$. That is, suppose $|\beta - \beta'| > c$ for all β, $\beta' \in \mathbb{C}$ such that $\beta \neq \beta'$ and β and β' are both roots of the norm. Suppose g has leading coefficient $g_m \in \mathbb{Q}[\alpha]$ and roots $\beta_1, \beta_2, \ldots, \beta_m$ (not necessarily distinct). If β is any root of g then clearly $g(\beta) = 0$. If β' is a root of the norm of g which is not also a root of g, then $|\beta - \beta_i| > c$ for $1 \le i \le m$; hence

$$\begin{aligned} |g(\beta')| &= |g_m \prod_{i=1}^{m} (\beta' - \beta_i)| \\ &= |g_m| \prod_{i=1}^{m} (|\beta' - \beta_i|) \\ &> |g_m| c^m. \end{aligned}$$

Using an efficient method for the isolation of roots of integer polynomials to obtain sufficiently good numerical estimates for the generator α of $\mathbb{Q}[\alpha]$ and for a root β of the norm of g over $\mathbb{Q}$, we can decide whether β is a root of g by computing $|g(\beta)|$ to within precision $(|g_m|c^m)/3$. We complete our description of an algorithm for root isolation over number fields by deriving lower bounds for $|g_m|$ and c, and showing that $g(\beta)$ can be estimated to the required precision in polynomial time.

We first consider the case that g is irreducible. Suppose, then, that

$$f = t^n + \sum_{i=0}^{n-1} f_i t^i \in \mathbb{Z}[t], \quad \text{and} \quad g = \sum_{j=0}^{m} g_j x^j = \sum_{j=0}^{m} \sum_{k=0}^{n-1} g_{j,\,k} \alpha^k x^j \in (\mathbb{Z}[\alpha])[x],$$

for f_i, $g_{j,\,k} \in \mathbb{Z}$, $g_j \in \mathbb{Z}[\alpha]$, such that $|f_i|$, $|g_{j,\,k}| < 2^M$ for some $M \in \mathbb{Z}$, and such that f is irreducible in $\mathbb{Q}[t]$ and g is irreducible in $(\mathbb{Q}[\alpha])[x]$. Suppose α is a root of f. We apply inequalities stated by Mignotte [89] to show that we can distinguish between roots of g, and other roots of $N_{\mathbb{Q}[\alpha]/\mathbb{Q}}(g)$, in polynomial time.

We first obtain bounds for the absolute value of α and for the absolute value of a root β of the norm of g over $\mathbb{Q}$. Applying Cauchy's inequality (Mignotte [89], Theorem 2 and corollary) to the polynomial f, we obtain the bounds

$$(2^M + 1)^{-1} < |\alpha| < 2^M + 1.$$

It follows that

$$|g_j| < 2^M \sum_{j=0}^{n-1} |\alpha^j| < 2^{n(M+1)} \quad \text{for } 0 \le j \le m.$$

We will also need a lower bound for nonzero coefficients g_j; each is a root of a monic polynomial $N_{\mathbb{Q}[\alpha]/\mathbb{Q}}(x - g_j)$, a polynomial of degree n in $\mathbb{Z}[x]$. Applying Proposition 1.4.14, we note that

$$N_{\mathbb{Q}[\alpha]/\mathbb{Q}}(x - g_j) = \operatorname{Res}_t(x - \sum_{k=0}^{n-1} g_{j,\,k} t^k,\, f),$$

the determinant of a $(2n-1) \times (2n-1)$ matrix of polynomials, each with degree at most 1 in x. Further, the entries of all but n columns of this matrix are integers. Using this expression, we obtain the upper bound

$$(2n-1)!\, 2^n 2^{M(2n-1)} < 2^{2n(M + \log(2n) + 1)}$$

for the absolute value of each coefficient of $N_{\mathbb{Q}[\alpha]/\mathbb{Q}}(x - g_j)$. Applying the corollary of Cauchy's inequality again, we conclude that

$$|g_j| > 2^{-2n(M + \log(2n) + 1) - 1} \quad \text{for all nonzero } g_j,$$

and that

$$2^{-3n(M+\log(2n)+1)} < |\beta| < 2^{3n(M+\log(2n)+1)}.$$

We next compute a lower bound for the root separation, c. Applying Theorem 4 of Mignotte, we observe that the squarefree part of $N_{\mathbb{Q}[\alpha]/\mathbb{Q}}(g)$ has degree at most mn, and coefficients whose absolute values have sum at most

$$2^{mn}(((mn+1)^{\frac{1}{2}})\,2^{2n(M+\log(2n)+1)}) < 2^{n(m+2M+3\log(mn)+5)}.$$

Applying the corollary of Theorem 5 of Mignotte, we conclude that

$$\begin{aligned} c &> \sqrt{3}\cdot(mn)^{-(mn+2)/2}\,2^{-n(m+2M+3\log(mn)+5)(mn-1)} \\ &> \sqrt{3}\cdot 2^{-mn^2(m+2M+4\log(mn)+5)}. \end{aligned}$$

We conclude that if β is a root of $N_{\mathbb{Q}[\alpha]/\mathbb{Q}}(g)$ which is not also a root of g, then

$$\begin{aligned} |g(\beta)| &> 2^{-2n(M+\log(2n)+1)-1}\,\sqrt{3}^{\,m}\cdot 2^{-m^2n^2(m+2M+4\log(mn)+5)} \\ &> \sqrt{3}^{\,m}\cdot 2^{-m^2n^2(m+3M+5\log(mn)+7)}. \end{aligned}$$

Hence we can decide whether β is a root of g by computing $|g(\beta)|$ to within accuracy ϵ, for

$$\begin{aligned} \epsilon &< \frac{1}{3}\,\sqrt{3}^{\,m}\cdot 2^{-m^2n^2(m+3M+5\log(mn)+7)} \\ &< \sqrt{3}^{\,m-2}\cdot 2^{-m^2n^2(m+3M+5\log(mn)+7)}. \end{aligned}$$

It remains only to show that $|g(\beta)|$ can be computed to this precision, using "easily computed" numerical estimates of α and β. Suppose now that we have computed estimates $\hat{\alpha}$ and $\hat{\beta}$ of α and β, respectively, with $|\hat{\alpha}-\alpha| < \delta$ and $|\hat{\beta}-\beta| < \delta$. The estimates can be used to decide whether $g(\beta)=0$ if

$$|g(\beta) - \sum_{j=0}^{m}\sum_{k=0}^{n-1} g_{j\,k}\hat{\alpha}^k\hat{\beta}^j| < \epsilon.$$

This is clearly the case if

$$\sum_{j=0}^{m}\sum_{k=0}^{n-1} |g_{j\,k}|\cdot|\hat{\alpha}^k\hat{\beta}^j - \alpha^k\beta^j| < \epsilon.$$

Using the upper bounds we obtained for $|g_{j\,k}|$, $|\alpha|$, and $|\beta|$, we see that the estimates can be used reliably if ϵ is larger than

$$\sum_{j=0}^{m} \sum_{k=0}^{n-1} 2^M \left((2^{M+1} + \delta)^k (2^{3n(M+\log(2n)+1)} + \delta)^j - (2^{M+1})^k (2^{3n(M+\log(2n)+1)})^j \right),$$

and it can be shown that this is true if

$$\begin{aligned} \delta &< 2^{-2nm(3M+2\log(nm)+4)}\,\epsilon \\ &< \sqrt{3}^{\,m-2} \cdot 2^{-m^2n^2(m+6M+7\log(mn)+11)}. \end{aligned}$$

We use these estimates to obtain the following algorithm.

Algorithm **Isolation of Roots via the Norm**

Input.
- Integers n, m, $M > 0$.
- The coefficients of a monic irreducible polynomial $f \in \mathbb{Z}[t]$
  $f = t^n + f_{n-1}t^{n-1} + \cdots + f_1 t + f_0$, of degree n,
  with $|f_i| < 2^M$ for $0 \le i < n$.
- Coordinates of an isolating rectangle for a root α of f.
- The coefficients g_m, g_{m-1}, …, g_1, g_0 of an irreducible polynomial
  $g = g_m x^m + g_{m-1}x^{m-1} + \cdots + g_1 x + g_0$ of degree m in $(\mathbb{Z}[\alpha])[x]$,
  with each coefficient g_i given by elements $g_{i,\,0}$, $g_{i,\,1}$, …, $g_{i,\,n-1}$ of $\mathbb{Z}$
  such that $|g_{i,\,j}| < 2^M$ for $0 \le j < n$ and such that
  $g_i = g_{i,\,n-1}\alpha^{n-1} + \cdots + g_{i,\,1}\alpha + g_{i,\,0}$.

Output.
- The coefficients of the minimal polynomial over $\mathbb{Q}$ of the roots of g.
- Coordinates of isolating rectangles for each of these roots.

(1) Compute the squarefree part, h, in $\mathbb{Q}[x]$, of
$N_{\mathbb{Q}[\alpha]/\mathbb{Q}}(g) = (-1)^{mn}\mathrm{Res}_t(\hat{g},\, f) \in \mathbb{Z}[t]$,
for $\hat{g} \in \mathbb{Z}[t, x]$ with degree less than n in t such that $\hat{g}(\alpha, x) = g$.
The polynomial h is the minimal polynomial in $\mathbb{Q}[x]$ of the roots of g.

(2) Use an asymptotically fast method for the isolation of roots of integer polynomials to estimate α and each of the roots of h, to precision
$\delta = \sqrt{3}^{\,m-2} \cdot 2^{-m^2n^2(m+6M+7\log(nm)+11)}$.

(3) For each root β of h, use the estimates computed in step 2 to compute an approximation of $|g(\beta)|$. Return β as a root of g if and only if this estimate of $|g(\beta)|$ is less than ϵ, for
$\epsilon = \sqrt{3}^{\,m-2} \cdot 2^{-m^2n^2(m+3M+5\log(nm)+7)}$.

The bounds δ and ϵ used in this algorithm have been derived using bounds on the size, and separation, of roots of a squarefree polynomial. These bounds are not known to be the best possible; any improvement in these bounds will yield a corresponding improvement in the error bounds required for estimates used by this algorithm — and hence in the time required by an algorithm using this approach.

We now consider the general case — that g is not necessarily irreducible. We first note that there exists an integer $b > 0$ such that $bf_n, bf_{n-1}, \ldots, bf_0 \in \mathbb{Z}$, for $f_n, f_{n-1}, \ldots, f_0$ the coefficients of the irreducible polynomial

$$f = f_n t^n + f_{n-1} t^{n-1} + \cdots + f_1 t + f_0 \in \mathbb{Q}[t].$$

In particular, we can take b to be the lowest common multiple of the denominators of these rational coefficients. Let $\hat{f}_i = bf_i$, for $0 \le i \le n$. If α is a root of f in some algebraic closure of $\mathbb{Q}$, then α is also a root of the polynomial

$$\hat{f} = \hat{f}_n t^n + \hat{f}_{n-1} t^{n-1} + \cdots + \hat{f}_1 t + \hat{f}_0 \in \mathbb{Z}[t],$$

which is also irreducible in $\mathbb{Q}[t]$. Now $\hat{f}_n \in \mathbb{Z}$ and $\hat{f}_n \ne 0$. Setting $\bar{\alpha} = \hat{f}_n \alpha$, we see that $\mathbb{Q}[\alpha] = \mathbb{Q}[\bar{\alpha}]$, and that $\bar{\alpha}$ is a root of the monic irreducible polynomial

$$\begin{aligned} \bar{f} &= t^n + \bar{f}_{n-1} t^{n-1} + \cdots + \bar{f}_1 t + \bar{f}_0 \\ &= t^n + \hat{f}_n \hat{f}_{n-1} t^{n-1} + \cdots + \hat{f}_n^{n-1} \hat{f}_1 t + \hat{f}_n^n \hat{f}_0 \in \mathbb{Z}[t]. \end{aligned}$$

The binary representations of the coefficients of $\bar{f}$ have length polynomial in the representations of the coefficients of f.

Given a polynomial $g \in (\mathbb{Q}[\alpha])[x]$, with $g = \sum_{j=0}^{m} \sum_{k=0}^{n-1} g_{j,\,k} \alpha^k x^j$, with $g_{j,\,k} \in \mathbb{Q}$ for $0 \le j \le m$ and $0 \le k < n$, it is clear that we can compute integers $\bar{g}_{j,\,k}$ such that the polynomial

$$\bar{g} = \sum_{j=0}^{m} \sum_{k=1}^{n-1} \bar{g}_{j,\,k} \bar{\alpha}^k x^j \in (\mathbb{Z}[\bar{\alpha}])[x]$$

is a (nonzero) integer multiple of g, and hence has the same roots as g. Again, the coefficients of $\bar{g}$ can be computed in polynomial time.

Finally, we can compute the squarefree decomposition of $\bar{g}$, and then compute the irreducible factors of $\bar{g}$ (using the method of Landau [76]) in polynomial time. By doing so, we reduce the general case of our problem to the special case; our algorithm can be applied to isolate the roots of the polynomial g, using time polynomial in the size of the (original) input. We summarise this in the following theorem.

Theorem 1.5.1. Given an irreducible polynomial $f \in \mathbb{Q}[t]$, an isolating rectangle for a root α of f, and the coefficients of a polynomial $g \in (\mathbb{Q}[\alpha])[x]$, we can compute the minimal polynomial over $\mathbb{Q}$ and an isolating rectangle for each of the roots of g, in polynomial time.

1.6 Factoring Polynomials over $\mathbb{R}$ and $\mathbb{C}$

We now consider the factorisation of polynomials over $\mathbb{R}$ and over $\mathbb{C}$. As before, we assume we are given a polynomial $g \in (\mathbb{Q}[\alpha])[x]$, for α a root of an irreducible polynomial $f \in \mathbb{Q}[t]$. As noted earlier, we can assume without loss of generality that f is a monic polynomial with integer coefficients (so that α is an *algebraic integer,* and that g is a squarefree polynomial with coefficients in the ring $\mathbb{Z}[\alpha]$. If we are working over $\mathbb{R}$, so that we are assuming g has real coefficients, then we assume α to be a real root of f. Since $\mathbb{C}$ is an algebraically closed field, every monic irreducible polynomial $h \in \mathbb{C}[x]$ has the form $x + \beta$, for $\beta \in \mathbb{C}$. Thus an irreducible factorisation of g over $\mathbb{C}$ has the form

$$g = c(x + \beta_1)(x + \beta_2) \cdots (x + \beta_m)$$

for distinct $\beta_1, \beta_2, \dots, \beta_m \in \mathbb{C}$, and for c the leading coefficient of g. Monic polynomials in $\mathbb{R}[x]$ which are irreducible over $\mathbb{R}$ have degree either one or two; thus an irreducible factorisation of g over $\mathbb{R}$ has the form

$$g = c(x + \beta_{1,\,0}) \cdots (x + \beta_{k,\,0})(x^2 + \beta_{k+1,\,1}x + \beta_{k+1,\,0}) \cdots (x^2 + \beta_{k+l,\,1}x + \beta_{k+l,\,0})$$

with all of these polynomials distinct, and with $m = k + 2l$.

Since the coefficients of g all lie in the number field $\mathbb{Q}[\alpha]$, the roots of g — and the coefficients of the irreducible (real or complex) factors of g — lie in some larger number field $\mathbb{Q}[\zeta]$, a *splitting field* for g. In principle, then, we could factor g (over $\mathbb{R}$ or $\mathbb{C}$) by computing a generator ζ for a splitting field of g (or, for factorisation over $\mathbb{R}$, for the largest real subfield of a splitting field), and then perform exact computations in this larger field. Landau [76] includes an algorithm for the computation of such a generator. However, there exist polynomials of degree n in $\mathbb{Q}[x]$ (and for arbitrary n) whose splitting fields all have degree at least $n!$ over $\mathbb{Q}$; we cannot compute (or even write down) the minimal polynomial over $\mathbb{Q}$ of a generator of such a splitting field using time polynomial in n.

We obtain a useful factorisation of g over $\mathbb{R}$ or $\mathbb{C}$ by using a more general representation of the splitting field. If g has factorisation

$$g = ch_1h_2 \cdots h_s$$

over $\mathbb{R}$ or $\mathbb{C}$, for c the leading coefficient of g, and for distinct monic irreducible polynomials $h_1, h_2, \dots, h_s$, then each irreducible factor h_i has coefficients in a number field $\mathbb{Q}[\zeta_i]$ which is a *small* (that is, polynomial degree) extension of $\mathbb{Q}[\alpha]$. Our "polynomial size" factorisation of g includes the minimal polynomial over $\mathbb{Q}$ and an isolating rectangle over $\mathbb{C}$ (for factorisation over $\mathbb{C}$) or isolating interval over $\mathbb{R}$ (for factorisation over $\mathbb{R}$), for each generator ζ_i. The coefficients of h_i will be represented as elements of $\mathbb{Q}[\zeta_i]$. We also give a representation of α as an element of $\mathbb{Q}[\zeta_i]$, in order to establish an embedding of $\mathbb{Q}[\alpha]$ within $\mathbb{Q}[\zeta_i]$.

We give a formal description of the computational problems "Factorisation over $\mathbb{R}$" and "Factorisation over $\mathbb{C}$" on the next two pages. We will show that each of these problems can be solved using time polynomial in n, m, and M.

We first consider the conceptually simpler problem, "Factorisation over $\mathbb{C}$". If g is irreducible in $(\mathbb{Q}[\alpha])[x]$, then the minimal polynomial of each root of g over $\mathbb{Q}$ is the squarefree part of $N_{\mathbb{Q}[\alpha]/\mathbb{Q}}(g)$. This polynomial, and isolating rectangles of each of the roots of g, are computed by the algorithm "Isolation of Roots via the Norm" discussed earlier. In the more general case that g is squarefree, we obtain these polynomials and isolating rectangles by factoring g in $(\mathbb{Q}[\alpha])[x]$ using the algorithm given by Landau [76], and then considering each irreducible factor of g (in $(\mathbb{Q}[\alpha])[x]$) separately. Hence we can compute minimal polynomials over $\mathbb{Q}$, and isolating rectangles in $\mathbb{C}$, for each root of a squarefree polynomial $g \in (\mathbb{Q}[\alpha])[x]$, using polynomial time.

It now remains to compute the minimal polynomial over $\mathbb{Q}$, and an isolating rectangle in $\mathbb{C}$, for a primitive generator ζ_i of the number field $\mathbb{Q}[\zeta_i] = \mathbb{Q}[\alpha, \beta_i]$, and to express α and β_i in terms of ζ_i, for $1 \leq i \leq m$. Loos [87] provides an efficient algorithm for this computation (namely, Algorithm 2 (SIMPLE)), for the case $\alpha, \beta_i \in \mathbb{R}$. It is a simple matter to check that the "interval arithmetic" used in this algorithm can be replaced by computations and refinements of isolating rectangles in $\mathbb{C}$ (Pinkert's method is sufficient for this), to generalise the algorithm so that it can be used for arbitrary algebraic numbers $\alpha, \beta_i \in \mathbb{C}$. Since the algorithm of Loos computes the remaining values specified as output for the problem "Factorisation over $\mathbb{C}$", we conclude that this problem can be solved using a polynomial number of operations over $\mathbb{Q}$.

It is easily checked that the values computed all have lengths polynomial in the input size, and that these computations can also be performed using a polynomial number of Boolean operations.

Problem **Factorisation over $\mathbb{R}$**

Input.

- Integers n, m, $M > 0$.
- The coefficients of a monic irreducible polynomial $f \in \mathbb{Z}[t]$
  $f = t^n + f_{n-1}t^{n-1} + \cdots + f_1 t + f_0$, of degree n,
  with $|f_i| < 2^M$ for $0 \le i < n$.
- Endpoints of an isolating interval of a real root α of f.
- The coefficients g_m, g_{m-1}, $\ldots$, g_1, g_0 of a squarefree polynomial
  $g = g_m x^m + g_{m-1}x^{m-1} + \cdots + g_1 x + g_0$ of degree m in $(\mathbb{Z}[\alpha])[x]$,
  with each coefficient g_i given by elements $g_{i,\,0}$, $g_{i,\,1}$, $\ldots$, $g_{i,\,n-1}$ of $\mathbb{Z}$ such that $|g_{i,\,j}| < 2^M$ for $0 \le j < n$, and such that
  $g_i = g_{i,\,n-1}\alpha^{n-1} + \cdots + g_{i,\,1}\alpha + g_{i,\,0}$.

Output.

- Integers k, $l \ge 0$ such that $k + 2l = m$, and such that
  $$g = g_m h_1 h_2 \cdots h_{k+l}$$
  is an irreducible factorisation of g over $\mathbb{R}$, for monic polynomials h_1, h_2, $\ldots$, h_k of degree 1, and h_{k+1}, h_{k+2}, $\ldots$, h_{k+l} of degree 2.
- The minimal polynomial over $\mathbb{Q}$ (with degree d_i) and an isolating interval in $\mathbb{R}$ for the algebraic integer $\zeta_i \in \mathbb{R}$, such that α and the coefficients of the polynomial h_i lie in $\mathbb{Q}[\zeta_i]$, for $1 \le i \le k + l$.
- For $1 \le i \le k$: numbers $a_{i,\,0}$, $a_{i,\,1}$, $\ldots$, $a_{i,\,d_i-1}$ and $b_{i,\,0}$, $b_{i,\,1}$, $\ldots$, $b_{i,\,d_i-1} \in \mathbb{Q}$ with
  $$\alpha = a_{i,\,0} + a_{i,\,1}\zeta_i + \cdots + a_{i,\,d_i-1}\zeta_i^{d_i-1}, \quad \text{and}$$
  $$\beta_{i,\,0} = b_{i,\,0} + b_{i,\,1}\zeta_i + \cdots + b_{i,\,d_i-1}\zeta_i^{d_i-1},$$
  for $\beta_{i,\,0} \in \mathbb{Q}[\zeta_i]$ such that $h_i = x + \beta_{i\,0}$ is the i^{th} irreducible polynomial in our factorisation of g over $\mathbb{R}$.
- For $k + 1 \le i \le k + l$: numbers $a_{i,\,0}$, $a_{i,\,1}$, $\ldots$, $a_{i,\,d_i-1}$, $b_{i,\,0}$, $b_{i,\,1}$, $\ldots$, $b_{i,\,d_i-1}$, and $c_{i,\,0}$, $c_{i,\,1}$, $\ldots$, $c_{i,\,d_i-1} \in \mathbb{Q}$ such that
  $$\alpha = a_{i,\,0} + a_{i,\,1}\zeta_i + \cdots + a_{i,\,d_i-1}\zeta_i^{d_i-1},$$
  $$\beta_{i,\,1} = b_{i,\,0} + b_{i,\,1}\zeta_i + \cdots + b_{i,\,d_i-1}\zeta_i^{d_i-1}, \quad \text{and}$$
  $$\beta_{i,\,0} = c_{i,\,0} + c_{i,\,1}\zeta_i + \cdots + c_{i,\,d_i-1}\zeta_i^{k_i-1},$$
  for $\beta_{i,\,1}$, $\beta_{i,\,0} \in \mathbb{Q}[\zeta_i]$ such that $h_i = x^2 + \beta_{i,\,1}x + \beta_{i,\,0}$ is the i^{th} irreducible polynomial in our factorisation of g over $\mathbb{R}$.

Problem **Factorisation over $\mathbb{C}$**

Input.

- Integers n, m, $M > 0$.
- The coefficients of a monic irreducible polynomial $f \in \mathbb{Z}[t]$
  $$f = t^n + f_{n-1}t^{n-1} + \cdots + f_1 t + f_0, \text{ of degree } n,$$
  with $|f_i| < 2^M$ for $1 \leq i < n$.
- Coordinates of an isolating rectangle of a root α of f.
- The coefficients $g_m, g_{m-1}, \ldots, g_1, g_0$ of a squarefree polynomial
  $$g = g_m x^m + g_{m-1}x^{m-1} + \cdots g_1 x + g_0 \text{ of degree } m \text{ in } (\mathbb{Z}[\alpha])[x]$$
  with each coefficient g_i given by elements $g_{i,0}, g_{i,1}, \ldots, g_{i,n-1}$ of $\mathbb{Z}$ such that $|g_{i,j}| < 2^M$ for $0 \leq j < n$ and such that
  $$g_i = g_{i,n-1}\alpha^{n-1} + \cdots + g_{i,1}\alpha + g_{i,0}.$$

Output.

- The minimal polynomial over $\mathbb{Q}$ (with degree d_i) and an isolating rectangle in $\mathbb{C}$ for algebraic integers $\zeta_1, \zeta_2, \ldots, \zeta_m \in \mathbb{C}$, such that α and the algebraic number β_i both lie in $\mathbb{Q}[\zeta_i]$, for $1 \leq i \leq m$, and for β_i such that
  $$h = g_m(x + \beta_1)(x + \beta_2)\cdots(x + \beta_m)$$
  is an irreducible factorisation of g over $\mathbb{C}$.
- Numbers $a_{i,0}, a_{i,1}, \ldots, a_{i,d_i-1}$ and $b_{i,0}, b_{i,1}, \ldots, b_{i,d_i-1} \in \mathbb{Q}$ such that
  $$\alpha = a_{i,0} + a_{i,1}\zeta_i + \cdots + a_{i,d_i-1}\zeta_i^{d_i-1},$$
  and
  $$\beta_i = b_{i,0} + b_{i,1}\zeta_i + \cdots + b_{i,d_i-1}\zeta_i^{d_i-1},$$
  for $1 \leq i \leq m$.

If our polynomial g splits completely into linear factors over $\mathbb{R}$, then the method sketched above can also be applied to compute the factorisation of g over $\mathbb{R}$. In general, Sturm sequences can be used to compute the number of real roots of g. Collins and Loos [26] include a description of this method. Since the polynomial g has real coefficients, the remaining roots occur in conjugate pairs, γ and $\bar{\gamma}$. Using the methods sketched above (for factorisation over $\mathbb{C}$), we can compute minimal polynomials over $\mathbb{Q}$ and isolating rectangles for each of the nonreal roots of g. Refining the isolating rectangles (if necessary), we can also match up each of the conjugate pairs of roots γ_i and $\bar{\gamma}_i$, for $l+1 \leq i \leq l+h$.

Since we are factoring g over $\mathbb{R}$, we are more interested in the (real) coefficients of the irreducible factor

$$x^2 + \beta_{i,\,1}x + \beta_{i,\,0}$$

of g having complex roots γ_i and $\bar{\gamma}_i$ than we are with the roots themselves. Using the equation

$$x^2 + \beta_{i,\,1}x + \beta_{i,\,0} = (x - \gamma_i)(x - \bar{\gamma}_i),$$

we see that

$$\beta_{i,\,1} = -(\gamma_i + \bar{\gamma}_i) \qquad \text{and} \qquad \beta_{i,\,0} = \gamma_i \cdot \bar{\gamma}_i.$$

It is clear that isolating intervals of $\beta_{i,\,1}$ and $\beta_{i,\,0}$ can be computed from isolating rectangles of γ_i and $\bar{\gamma}_i$, provided that the minimal polynomials of $\beta_{i,\,1}$ and $\beta_{i,\,0}$ can be computed. We compute these polynomials from the minimal polynomials of γ_i and $\bar{\gamma}_i$, by computing several resultants of polynomials, using the following relationships.

Proposition 1.6.1. (Loos.) Let $A = a_m \prod_{i=1}^{m}(x - \alpha_i)$ and $B = b_n \prod_{j=1}^{n}(x - \beta_j)$ be polynomials with positive degree in $R[x]$, for an integral domain R, with roots $\alpha_1, \alpha_2, \ldots, \alpha_m$ and $\beta_1, \beta_2, \ldots, \beta_n$ respectively.

(i) The polynomial $\mathrm{Res}_y(A(x-y),\, B(y))$ has roots $\gamma_{i\,j} = \alpha_i + \beta_j$, for $1 \leq i \leq m$, $1 \leq j \leq n$.

(ii) The polynomial $\mathrm{Res}_y(A(x+y),\, B(y))$ has roots $\gamma_{i\,j} = \alpha_i - \beta_j$, for $1 \leq i \leq m$, $1 \leq j \leq n$.

(iii) The polynomial $\mathrm{Res}_y(y^m A(x/y),\, B(y))$ has roots $\gamma_{i\,j} = \alpha_i \cdot \beta_j$, for $1 \leq i \leq m$, $1 \leq j \leq n$.

(iv) The polynomial $\mathrm{Res}_y(A(xy),\, B(y))$ has roots $\gamma_{i\,j} = \alpha_i / \beta_j$, for $1 \leq i \leq m$, $1 \leq j \leq n$.

This is a restatement of Theorem 6 of Loos [87]. It can be used to compute polynomials in $\mathbb{Q}[x]$ having $\beta_{i,\,1}$ and $\beta_{i,\,0}$ as roots; we obtain the minimal polynomials of these values over $\mathbb{Q}$ using factorisation over $\mathbb{Q}$, and comparisons of isolating rectangles. We then use this information to compute the minimal polynomial over $\mathbb{Q}$, and an isolating interval in $\mathbb{R}$, of a primitive generator ζ_i of $\mathbb{Q}[\zeta_i] = \mathbb{Q}[\alpha,\, \beta_{i,\,1},\, \beta_{i,\,0}]$, and express α, $\beta_{i,\,1}$, and $\beta_{i,\,0}$ in terms of ζ_i, as sketched in the discussion of factorisation over $\mathbb{C}$.

Again, it is straightforward to check that the algorithms of Loos can be used to compute these values using a polynomial number of Boolean operations.

Theorem 1.6.2. The problems "Factorisation over $\mathbb{R}$" and "Factorisation over $\mathbb{C}$" can each be solved using a number of Boolean operations polynomial in n, m, and N.

Thus we can factor polynomials over number fields, and we can factor polynomials with algebraic numbers as coefficients over $\mathbb{R}$ or $\mathbb{C}$, in polynomial time. We will apply these results in later sections to show that a number of problems involving decompositions of associative algebras and of linear representations of groups can be solved in polynomial time, as well.

2. Computations for Associative Algebras

In this section we discuss computations for finite-dimensional associative algebras over a field F. The structure of these algebras is well understood; the goal of this section is to find algorithms which can be used to decompose algebras in the manner described by the classical structure theorems for rings (and, in particular, algebras).

Friedl and Rónyai ([43]) and Rónyai ([102]–[104]) have obtained algorithms for the decomposition of algebras over $\mathbb{Q}$ and finite fields, and evidence that one stage of this decomposition is difficult. We review their algorithms, and use their techniques to obtain arithmetic reductions from these computational problems for algebras to problems concerning factorisation of polynomials. We also give some new algorithms for these computations (in particular, see Sections 2.4.3 and 2.4.4), and apply the methods of Friedl and Rónyai, and other existing techniques, to decompose algebras over $\mathbb{R}$ and $\mathbb{C}$.

In Section 2.1 we review the classical ("Wedderburn") structure theorems for associative algebras. In Section 2.2, we give additional material needed for us to define computational problems corresponding to these theorems. We include a standard representation of an arbitrary finite-dimensional associative algebra over a field F as a matrix algebra over F, and discuss the cost of obtaining this "concrete" representation from more general representations of associative algebras over F. With this matrix representation in mind, we define computational problems corresponding to the structure theorems, which take as input a basis (of matrices) over F for a matrix algebra, and return bases for components of this algebra. The structure theorems describe three distinct phases in the decomposition of a finite-dimensional associative algebra; algorithms for the three computational problems corresponding to these phases are discussed in Sections 2.3, 2.4, and 2.5 respectively. We indicate some directions for further research in Section 2.6.

2.1. Definitions and Notation: The Structure Theorems

We begin with definitions leading to the statement of the structure theorems for associative algebras over a field. The material presented here is standard. For more comprehensive treatments of this see (for example) the texts of Curtis and Reiner [31], Jacobson [67], [68], or van der Waerden [117], [118]. In general, we adopt the notation of Curtis and Reiner [31].

Henceforth F denotes a field, and A denotes an associative algebra over F.

Definition 2.1.1. An *associative algebra A over a field F* is a ring with an identity element which is at the same time a vector space over F, such that the scalar multiplication in the vector space and the ring multiplication satisfy the axiom

$$\alpha(ab) = (\alpha a)b = a(\alpha b) \qquad \text{for } \alpha \in F \text{ and } a, b \in A.$$

A subring of A which is also an F-subspace of A is called a *subalgebra* of A.

We will restrict attention to algebras which are finite-dimensional (as vector spaces) over F. We note some examples of associative algebras (which we will be discussing further) below.

Example 2.1.2. F is an associative algebra (of dimension 1) over itself. If $E \supseteq F$ is a finite algebraic extension of F, then E is an associative algebra over F.

Example 2.1.3. Let $F = \mathbb{R}$, and let $A = \mathbb{H}$, the ring of real quaternions. $\mathbb{H}$ is a vector space of dimension 4 over $\mathbb{R}$, with basis $\{ 1, i, j, k \}$ over $\mathbb{R}$, where

$$i^2 = j^2 = k^2 = -1, \quad ij = -ji = k, \quad jk = -kj = i, \quad \text{and} \quad ki = -ik = j.$$

$\mathbb{H}$ is a (noncommutative) associative algebra over $\mathbb{R}$.

Example 2.1.4. Let $n > 0$. Any subring of the ring of $n \times n$ matrices over F which includes the ring of *homotheties* $\{ \alpha I_n : \alpha \in F \}$ is an algebra over F. In particular, the ring of upper triangular $n \times n$ matrices over F, and the ring of all $n \times n$ matrices over F, are both associative algebras over F.

We will see later that every associative algebra of dimension n over a field F is isomorphic to a subalgebra of the algebra of $n \times n$ matrices over F.

Example 2.1.5. Let $f \in F[x]$ be a polynomial with degree $n \geq 0$; the ring $A = F[x]/(f)$ is an associative algebra of dimension n over F.

Example 2.1.6. Let $G = \{ g_1, g_2, \ldots, g_n \}$ be a finite group. The *group algebra*, FG, is the set of formal linear combinations

$$\{ \alpha_1 g_1 + \alpha_2 g_2 + \cdots + \alpha_n g_n : \alpha_1, \alpha_2, \cdots, \alpha_n \in F \}.$$

Two linear combinations are considered to be equal if and only if their coefficients are the same. Addition is straightforward, and multiplication is defined using the group operation:

$$\Big(\sum_{i=1}^{n} \alpha_i g_i\Big) + \Big(\sum_{i=1}^{n} \beta_i g_i\Big) = \sum_{i=1}^{n} (\alpha_i + \beta_i) g_i;$$

$$\Big(\sum_{i=1}^{n} \alpha_i g_i\Big) \cdot \Big(\sum_{i=1}^{n} \beta_i g_i\Big) = \sum_{1 \le i,\, j \le n} \alpha_i \beta_j g_i g_j = \sum_{i=1}^{n} \gamma_i g_i,$$

where

$$\gamma_i = \sum_{\substack{1 \le j,\, k \le n \\ g_j g_k = g_i}} \alpha_j \beta_k.$$

FG is an associative algebra of dimension $n = |G|$ over F.

Our definition of associative algebra is not the most general possible; some authors (including Friedl and Rónyai [43]) drop the condition that A include a multiplicative identity. Consider, for example, the ring of strictly upper triangular $n \times n$ matrices over a field F:

$$A = \{\, U = (U_{i\,j})_{1 \le i,j \le n} \in M_{n \times n}(F) \,:\, U_{i\,j} = 0 \text{ if } j \le i \,\}.$$

A is an "associative algebra over F" according to the definition used by Friedl and Rónyai, but not according to Definition 2.1.1. Most of the results which follow apply for either definition; we will note instances where the choice of definition is important. (See, in particular, Example 2.1.7 below, Example 2.2.7 and the remarks preceding it, and Section 2.3.3.)

Example 2.1.7. Let $\hat{A}$ be a vector space of dimension $n > 0$ over F which satisfies the conditions of Definition 2.1.1, except that $\hat{A}$ does not include a multiplicative identity (so $\hat{A}$ is an associative algebra over F, according to the definition of Friedl and Rónyai). Consider the set

$$A = \{\, \alpha 1_A + a \,:\, \alpha \in F,\, a \in \hat{A} \,\},$$

with $1_A \notin \hat{A}$ and with addition and multiplication in A defined by

$$(\alpha 1_A + a) + (\beta 1_A + b) = (\alpha + \beta) 1_A + (a + b),$$

$$(\alpha 1_A + a) \cdot (\beta 1_A + b) = (\alpha\beta) 1_A + (\alpha b + \beta a + ab),$$

for $\alpha, \beta \in F$ and $a, b \in \hat{A}$. Then A is an associative algebra over F (according to Definition 2.1.1) with multiplicative identity 1_A, and of dimension $n + 1$ over F.

Associative algebras can be classified according to the types of ideals they include (as rings). Suppose I and J are left ideals of A. We denote by $I+J$, $I \cdot J$, and I^m the following sets:

(i) $I+J = \{\, a+b \,:\, a \in I,\, b \in J \,\}$;

(ii) $I \cdot J$ is the smallest left ideal containing the set $\{\, ab \,:\, a \in I$ and $b \in J \,\}$;

(iii) I^m is the smallest left ideal containing the set of products $\{\, a_1 a_2 \cdots a_m \,:\, a_i \in I$ for $1 \leq i \leq m \,\}$.

Inductively, $I^1 = I$, and $I^{n+1} = I^n \cdot I$ for $n > 0$. We define right (respectively, two-sided) ideals $I+J$, $I \cdot J$, and I^m for right (respectively, two-sided) ideals I and J in a similar way.

Definition 2.1.8. Let A be an associative algebra over a field F. An element a of A is *nilpotent* if $a^n = 0$ for some $n \geq 0$. An element a of A is *strongly nilpotent* if ab is nilpotent for all $b \in A$. An ideal I of A is *nilpotent* if $I^n = 0$ for some $n > 0$.

Example 2.1.9. Let A be the algebra of 2×2 matrices over the field F. The element

$$a = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \in A$$

is nilpotent, since $a^2 = 0$. However,

$$a \cdot \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, \qquad \text{and} \qquad \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}^n = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \neq 0$$

so a is not strongly nilpotent.

Example 2.1.10. Let A be the algebra of 2×2 upper triangular matrices over F,

$$A = \left\{ \begin{bmatrix} \alpha & \beta \\ 0 & \gamma \end{bmatrix} \,:\, \alpha,\, \beta,\, \gamma \in F \right\},$$

and let

$$a = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \in A.$$

Then a is strongly nilpotent in A, since

$$\left(a \cdot \begin{bmatrix} \alpha & \beta \\ 0 & \gamma \end{bmatrix} \right)^2 = \begin{bmatrix} 0 & \gamma \\ 0 & 0 \end{bmatrix}^2 = 0$$

for all α, β, $\gamma \in F$. The ideal

$$I = \left\{ \begin{bmatrix} 0 & \alpha \\ 0 & 0 \end{bmatrix} \; : \; \alpha \in F \right\}$$

is a nilpotent ideal in A.

Definition 2.1.11. Let A be a finite-dimensional associative algebra over a field F. The *radical* of A, $\mathrm{rad}(A)$, is the sum of all nilpotent left ideals of A.

Example 2.1.12. Let $f \in F[x]$ be a polynomial with degree $n \geq 0$, $A = F[x]/(f)$, and let $g \in F[x]$ be the squarefree part of f. Then the radical of A is

$$\mathrm{rad}(A) = (g)/(f) = \{\, \alpha \cdot (g \bmod f) \; : \; \alpha \in A \,\}.$$

In particular, $\mathrm{rad}(A) = (0)$ if and only if f is squarefree.

See Section 2.3 for more examples of the radicals of associative algebras.

Since a left ideal of a finite-dimensional associative algebra A is also an F-subspace of A, it is clear that the radical of A is a subspace of A, as well as a left ideal of A ($\mathrm{rad}(A) = (0)$ if A has no nilpotent left ideals). In fact, more can be said about the structure of the radical of A.

Theorem 2.1.13. Let A be a finite-dimensional associative algebra over a field F. Then $\mathrm{rad}(A)$ is a two-sided nilpotent ideal, which contains every nilpotent left ideal of A, as well as every nilpotent right ideal of A.

See Curtis and Reiner [31], pp. 161–162, for a proof of this result for a class of rings which includes any finite-dimensional associative algebra over a field.

The definition given here for the radical of a finite-dimensional associative algebra A is not the only one used. Note, in particular, that Friedl and Rónyai [43] define the radical of A to be the set of all strongly nilpotent elements. Since we wish to apply their results, we will show that the definitions are equivalent: Suppose a is a strongly nilpotent element of A; then every element of the left ideal I generated by a is nilpotent. In fact, the ideal I is itself nilpotent: $I^m = (0)$ for some $m > 0$ (see Curtis and Reiner [31], page 160, for a proof). Hence $I \subseteq \mathrm{rad}(A)$; thus $a \in \mathrm{rad}(A)$, and the radical (as we define it) contains every strongly nilpotent element. On the other hand, if a is an element of the radical of A, and $b \in A$, then ab is in the radical (since this is an ideal) and ab is nilpotent (since the radical is a nilpotent ideal, by Theorem 2.1.13). Thus a is strongly nilpotent. It is clear, then, that the definitions of $\mathrm{rad}(A)$ are equivalent.

Since the radical of an associative algebra A is a two-sided ideal of A, as well as an F-subspace, it is clear that the *factor algebra*

$$A/\mathrm{rad}(A) = \{\, a + \mathrm{rad}(A) \,:\, a \in A \,\}$$

is itself a finite-dimensional associative algebra over F. It is also clear that $A/\mathrm{rad}(A)$ has radical (0). That is, $A/\mathrm{rad}(A)$ is semi-simple, as defined below.

Definition 2.1.14. A finite-dimensional associative algebra A is *semi-simple* if $\mathrm{rad}(A) = (0)$.

In Example 2.1.12 we noted that if $f \in F[x]$ then the algebra $A = F[x]/(f)$ is semi-simple if and only if f is squarefree.

It is clear that a finite-dimensional associative algebra A is semi-simple if and only if it has no nilpotent (left, right, or two-sided) ideals. We define more restrictive classes of algebras by considering their two-sided, and their one-sided, ideals.

Definition 2.1.15. A finite-dimensional associative algebra A over a field F is *simple* if the only two-sided ideals of A are A and (0).

Definition 2.1.16. A finite-dimensional associative algebra A over a field F is a *division algebra over F* if the only left ideals of A are A and (0).

Proposition 2.1.17 motivates the choice of name “division algebra”.

Proposition 2.1.17. Let A be a finite-dimensional associative algebra over a field F; then the following are equivalent.

(i) A is a division algebra over F.

(ii) The only left ideals of A are A and (0).

(iii) The only right ideals of A are A and (0).

(iv) A is semi-simple, and the only idempotent elements of A are 0 and 1.

(v) If $u \in A$ and $u \neq 0$ then u is a unit: There exists $v \in A$ such that $uv = vu = 1$.

(vi) A is a skew field with F in its centre.

Most of these implications are straightforward; the most difficult is the implication “(iv) $\Rightarrow$ (ii)”. For a proof of this, see Curtis and Reiner [31], pp. 160–161.

Example 2.1.18. If $f \in F[x]$ is irreducible then $A = F[x]/(f)$ is a simple algebra, and a division algebra, over F.

Example 2.1.19. The algebra $M_{n\times n}(F)$ of $n \times n$ matrices over F is a simple algebra for all $n > 0$. The algebra is a division algebra if and only if $n = 1$.

Example 2.1.20. The ring $\mathbb{H}$ of quaternions (defined in Example 2.1.3) is a noncommutative division algebra over $\mathbb{R}$.

Clearly, every division algebra is simple, and every simple algebra is semi-simple. The structure theorems stated below imply that semi-simple algebras can be decomposed into simple algebras, and that simple algebras can also be related to division algebras.

Suppose now that L_1 and L_2 are left ideals in a ring R.

Definition 2.1.21. L_1 and L_2 are *isomorphic* in R if there is a bijection ϕ from L_1 to L_2 such that $\phi(l_1 + l_2) = \phi(l_1) + \phi(l_2)$, and $\phi(rl) = r\phi(l)$, for all $r \in R$ and l, l_1, $l_2 \in L_1$.

Definition 2.1.22. L is *minimal*, or *irreducible*, in R, if $l \neq (0)$ and the only left ideal strictly contained in L (as a set) is (0).

Isomorphic and irreducible right or two-sided ideals in R are defined in an analogous way.

Theorem 2.1.23. *Let A be a semi-simple algebra over F and let L be a minimal nonzero left ideal of A. The sum B_L of all the minimal left ideals of A which are isomorphic to L is a simple algebra over F and a two-sided ideal of A. Furthermore, A is the direct sum of all the two-sided ideals B_L obtained by letting L range over a full set of non-isomorphic minimal left ideals of A.*

If A is finite-dimensional and semi-simple over F, the direct sum mentioned in the above theorem is finite: $A = B_1 \oplus B_2 \oplus \cdots \oplus B_m$ for two-sided ideals (and simple algebras) $B_1, B_2, \ldots, B_m$. (Note that B_i is not a subalgebra of A unless $m = 1$, since B_i does not include the multiplicative identity of A.)

Definition 2.1.24. The ideals $B_1, B_2, \ldots, B_m$ in the above summation are the *simple components of* A.

The decomposition of A into simple components is unique: for if A is a finite-dimensional semi-simple algebra and

$$A = B_1 \oplus B_2 \oplus \cdots \oplus B_m = C_1 \oplus C_2 \oplus \cdots \oplus C_l,$$

then $m = l$ and (after suitable reordering of the C_i's) $B_i = C_i$ for $1 \leq i \leq m$. Every two-sided ideal of A is the direct sum of a subset of the simple components of A.

Further, there exist idempotents $b_1, b_2, \ldots, b_m$ in A such that

$$b_1 + b_2 + \cdots + b_m = 1, \qquad b_i b_j = \delta_{i\,j} b_i \quad \text{for} \quad 1 \leq i, j \leq m, \qquad b_i \in B_i,$$

and such that $B_i = b_i A$. (Here, $\delta_{i\,j}$ is the *Kronecker delta:* $\delta_{i\,j} = 1$ if $i = j$, and $\delta_{i\,j} = 0$ otherwise.) Each b_i is in the *centre* of A; that is, b_i is an element of the set

$$\text{Centre}(A) = \{\, c \in A \;:\; ca = ac \text{ for all } a \in A \,\}.$$

Finally, b_i is the multiplicative identity of the simple algebra B_i.

For a proof of Theorem 2.1.23 and the above remarks, see Section 25 of Curtis and Reiner [31].

Theorem 2.1.25. (Wedderburn-Artin). Let A be a finite-dimensional simple algebra over a field F. Then for some $k > 0$, A is isomorphic to $M_{k \times k}(D)$, the ring of $k \times k$ matrices over D, for some finite-dimensional division algebra D over F. There exist minimal left ideals $L_1, L_2, \ldots, L_k$ of A which are each isomorphic to D^k, such that $A = L_1 \oplus L_2 \oplus \cdots \oplus L_k$. This decomposition is unique (only) up to isomorphism.

See Section 26 of Curtis and Reiner [31] for a proof of this result.

Taken together, Theorems 2.1.13, 2.1.23, and 2.1.25 comprise a structure theory for the finite-dimensional associative algebras over a field: Every finite-dimensional associative algebra A has a unique maximal nilpotent ideal, $\text{rad}(A)$; the factor algebra $A/\text{rad}(A)$ is semi-simple. Every finite-dimensional semi-simple algebra can be expressed as a direct sum of simple algebras. Finally, every finite-dimensional simple algebra is isomorphic to a ring of $k \times k$ matrices over a division algebra D, for some $k > 0$. In the following sections we discuss representations of algebras (as inputs and outputs for computational problems), and consider computational problems (the "Wedderburn decomposition" of an algebra) which correspond to these theorems.

2.2. Representations of Algebras for Computations

In this section we describe the method to be used to specify finite-dimensional associative algebras as inputs and outputs for computational problems — in particular, for problems corresponding to the decomposition of an associative algebra described in Section 2.1.

The matrix representation for an algebra given below (in Definition 2.2.2) is standard. One such representation is obtained for each basis for the algebra over the ground field. We will see in later sections that we can decompose an associative algebra by choosing a different basis for the algebra — one which isolates the algebra's components. Accordingly, we consider the problem of converting between the representations corresponding to two different bases for an algebra.

We also consider the problem of computing our standard representation of a (matrix) algebra A from a set of matrices which generate A under addition and multiplication. We show that there is an efficient (parallel) algorithm for this computation — see, in particular, Theorem 2.2.10. We will use this in Section 3 to obtain reductions between problems for matrix algebras and problems for matrix representations of groups.

Finally, we introduce the (standard) techniques we use to represent associative algebras over $\mathbb{R}$ and $\mathbb{C}$ using a set of constants in a number field — so that we can discuss Boolean algorithms for the decomposition of these algebras.

2.2.1. Regular Matrix Representations

In general, we use *regular matrix representations*, as defined below, to describe finite-dimensional associative algebras over a field.

Definition 2.2.1. Suppose A is an associative algebra of dimension n over a field F, and let $\{ a_1, a_2, \ldots, a_n \}$ be a basis for A over F. The *structure constants* for A with respect to this basis are the constants $\gamma_{i\,j\,k} \in F$ such that

$$a_i \cdot a_j = \sum_{k=1}^{n} \gamma_{i\,j\,k}\, a_k \qquad \text{for } 1 \le i,\ j,\ k \le n.$$

Definition 2.2.2. Let A, n, F, $\{ a_1, a_2, \ldots, a_n \}$, and $\gamma_{i\,j\,k}$ be as above. For $1 \le i \le n$, let $M_i \in M_{n\times n}(F)$ such that the $(j,\, k)^{\text{th}}$ entry of M_i is $\gamma_{i\,k\,j}$ for $1 \le j, k \le n$. That is,

$$M_i = \begin{bmatrix} \gamma_{i\,1\,1} & \gamma_{i\,2\,1} & \cdots & \gamma_{i\,n\,1} \\ \gamma_{i\,1\,2} & \gamma_{i\,2\,2} & \cdots & \gamma_{i\,n\,2} \\ \vdots & \vdots & \ddots & \vdots \\ \gamma_{i\,1\,n} & \gamma_{i\,2\,n} & \cdots & \gamma_{i\,n\,n} \end{bmatrix} \in M_{n\times n}(F).$$

Let $\phi : A \to M_{n\times n}(F)$ such that

$$\phi(\alpha_1 a_1 + \alpha_2 a_2 + \cdots + \alpha_n a_n) = \alpha_1 M_1 + \alpha_2 M_2 + \cdots + \alpha_n M_n$$

for $\alpha_1, \alpha_2, \ldots, \alpha_n \in F$. The representation $\phi(A)$ of A as a set of $n \times n$ matrices over F is called the *regular matrix representation* for A with respect to the basis $\{ a_1, a_2, \ldots, a_n \}$.

Proposition 2.2.3. The map $\phi : A \to \phi(A) \subseteq M_{n\times n}(F)$ is an algebra isomorphism.

The proof of Proposition 2.2.3 is straightforward: It is clear that $\phi(a+b) = \phi(a) + \phi(b)$ for all $a, b \in A$. Using the fact that multiplication in A is associative (in particular, that $(a_i \cdot a_j) \cdot a_h = a_i \cdot (a_j \cdot a_h)$) we check that $\sum_{h=1}^{n} \gamma_{i\,j\,h}\gamma_{h\,k\,l} = \sum_{h=1}^{n} \gamma_{i\,h\,l}\gamma_{j\,k\,h}$ for $1 \le i, j, k, l \le n$. It follows directly from this that $\phi(a_i \cdot a_j) = \phi(a_i) \cdot \phi(a_j)$ for $1 \le i, j \le n$; using linearity, we conclude that $\phi(a \cdot b) = \phi(a) \cdot \phi(b)$ for all $a, b \in A$. It is also clear from the definition of structure constants that $\phi(0)$ and $\phi(1)$ are respectively the zero and identity matrices in $M_{n\times n}(F)$. Finally, we use the fact that $a_1, a_2, \ldots, a_n$ is a basis for A over F to verify that the map ϕ is injective, as required to complete the proof.

It is clear that the matrices $\phi(a_1), \phi(a_2), \ldots, \phi(a_n)$ can be computed from the structure constants for the basis $a_1, a_2, \ldots, a_n$ for A, using time $O(n^3)$, linear in the size of the set of structure constants, or using arithmetic-Boolean circuits over F of size $O(n^3)$ and constant depth.

We continue with matrix representations of some algebras to be discussed later.

Example 2.2.4. Let $F = \mathbb{R}$, $A = \mathbb{H}$, and consider the basis $\{ 1, i, j, k \}$ discussed in Example 2.1.3. The regular representation of A with respect to this basis is given by

$$\phi(1) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \qquad \phi(i) = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \end{bmatrix},$$

$$\phi(j) = \begin{bmatrix} 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{bmatrix}, \qquad \phi(k) = \begin{bmatrix} 0 & 0 & 0 & -1 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}.$$

Example 2.2.5. Let G be a finite group $\{ g_1, g_2, \ldots, g_n \}$ and consider the group algebra FG discussed in Example 2.1.6. The elements of G comprise a basis for FG over F. The regular matrix representation $\phi : FG \to M_{n\times n}(F)$ with respect to this basis is given by

$$\phi(g_i)_{j\,k} = \begin{cases} 1 & \text{if } g_i \cdot g_k = g_j, \\ 0 & \text{otherwise.} \end{cases} \qquad \text{for } 1 \le i,\, j,\, k \le n.$$

Thus $\phi(g)$ is a permutation matrix for each $g \in G$, and the set

$$\phi(G) = \{ \phi(g) : g \in G \}$$

comprises a group of $n \times n$ matrices which is isomorphic to G.

Example 2.2.6. Suppose $f \in F[x]$ is a monic polynomial of degree n,

$$f = x^n + \alpha_{n-1}x^{n-1} + \alpha_{n-2}x^{n-2} + \cdots + \alpha_1 x + \alpha_0$$

for $\alpha_{n-1}, \alpha_{n-2}, \ldots, \alpha_1, \alpha_0 \in F$. Consider the algebra $A = F[x]/(f)$ discussed in Example 2.1.5. The elements

$$\{ 1 + (f),\, x + (f),\, x^2 + (f),\, \ldots,\, x^{n-1} + (f) \}$$

comprise a basis for A. The regular representation $\phi : A \to M_{n\times n}(F)$ of A with respect to this basis is given by

$$\phi(x^i + (f)) = \phi(x + (f))^i = \begin{bmatrix} 0 & 0 & \cdots & 0 & 0 & -\alpha_0 \\ 1 & 0 & \cdots & 0 & 0 & -\alpha_1 \\ 0 & 1 & \cdots & 0 & 0 & -\alpha_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 & -\alpha_{n-2} \\ 0 & 0 & \cdots & 0 & 1 & -\alpha_{n-1} \end{bmatrix}^i \in M_{n\times n}(F).$$

In particular, $\phi(x + (f))$ is the *companion matrix* of f.

Suppose now that $\hat{A}$ is an "associative algebra" of dimension n over F, as defined by Friedl and Rónyai, and that $\hat{A}$ does not include a multiplicative identity. As shown in Example 2.1.7, we can embed $\hat{A}$ in an associative algebra A which has dimension $n + 1$ over F. We obtain a regular matrix representation for $\hat{A}$ with respect to some basis $\{ a_1, a_2, \ldots, a_n \}$ by forming the regular matrix representation $\phi : A \to M_{(n+1)\times(n+1)}(F)$ of A with respect to the basis $\{ 1, a_1, a_2, \ldots, a_n \}$ of A, then restricting the domain to obtain a map $\hat{\phi} : \hat{A} \to M_{(n+1)\times(n+1)}(F)$. Thus we obtain a map taking elements of $\hat{A}$ to matrices of order $n + 1$, one more than the dimension of $\hat{A}$ over F.

Example 2.2.7. To see that this increase of order is necessary, consider the "associative algebra"

$$\hat{A} = \{ \alpha e \, : \, \alpha \in F \}$$

for $e \neq 0$, $e^2 = 0$. We obtain a "regular representation" $\hat{\phi} : \hat{A} \to M_{2\times 2}(F)$ given by

$$\hat{\phi}(e) = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}.$$

Since $e^2 = 0$, it is clear that the only "algebra homomorphism" $\psi : \hat{A} \to M_{1\times 1}(F)$ must map e^2, and e, to 0. Thus there are no "algebra isomorphisms" from $\hat{A}$ to $M_{1\times 1}(F)$.

2.2.2. Conversion Between Representations

We will be considering computational problems for finite-dimensional associative algebras corresponding to the "structure theorems" (Theorems 2.1.12, 2.1.23, and 2.1.25) of Section 2.1. In general, each problem will take as input the regular representation of an associative algebra A with respect to a basis $\{ a_1, a_2, \ldots, a_n \}$ over F, and will compute as output a second basis $\{ b_1, b_2, \ldots, b_n \}$ over F (which gives more information about the structure of A). Thus it will be useful to study the relationship between regular representations of an algebra A with respect to different bases.

Suppose now that A is a finite-dimensional associative algebra of dimension n over F, with bases $\{ a_1, a_2, \ldots, a_n \}$ and $\{ b_1, b_2, \ldots, b_n \}$ over F, and that

$$b_i = \sum_{j=1}^{n} \mu_{i\,j}\, a_j \qquad \text{for } \mu_{i\,j} \in F, \quad 1 \leq i, \, j \leq n.$$

Let $X \in M_{n\times n}(F)$ with $X_{i\,j} = \mu_{j\,i}$; then it is easily verified that if

$$a = \alpha_1 a_1 + \alpha_2 a_2 + \cdots + \alpha_n a_n = \beta_1 b_1 + \beta_2 b_2 \cdots + \beta_n b_n,$$

for $\alpha_1, \alpha_2, \ldots, \alpha_n, \beta_1, \beta_2, \ldots, \beta_n \in F$, then

$$\begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_n \end{bmatrix} = X \cdot \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_n \end{bmatrix}.$$

Suppose also that $\{\gamma_{i\,j\,k}\}$ and $\{\zeta_{i\,j\,k}\}$ are sets of structure constants for A with respect to the bases $\{a_1, a_2, \ldots, a_n\}$ and $\{b_1, b_2, \ldots, b_n\}$, respectively, so that

$$a_i \cdot a_j = \sum_{k=1}^{n} \gamma_{i\,j\,k}\, a_k \qquad \text{and} \qquad b_i \cdot b_j = \sum_{k=1}^{n} \zeta_{i\,j\,k}\, b_k \qquad \text{for } 1 \leq i,\, j,\, k \leq n.$$

We obtain regular representations ϕ and ψ for A with respect to these bases:

$$\phi(a_i)_{j\,k} = \gamma_{i\,k\,j} \quad \text{and} \quad \psi(b_i)_{j\,k} = \zeta_{i\,k\,j} \quad \text{for } 1 \leq i,\, j,\, k \leq n.$$

We can use the matrix X to convert from one representation to the other.

Proposition 2.2.8. Let A, n, $\{a_1, a_2, \ldots, a_n\}$, $\{b_1, b_2, \ldots, b_n\}$, X, ϕ, and ψ be as described above. Then the matrix X is nonsingular, and

$$\psi(a) = X^{-1}\phi(a)X \qquad \text{for all } a \in A.$$

Again, this result is both well known and easily proved. Since the matrix X has full rank, it is clear that we can prove it by verifying that $X \cdot \psi(b_i) = \phi(b_i) \cdot X$ for $1 \leq i \leq n$. It is easily checked (using the fact that $b_i \cdot b_j = \sum_{r=1}^{n} \sum_{s=1}^{n} \mu_{i\,r}\mu_{j\,s} a_r \cdot a_s$) that $\sum_{t=1}^{n} \zeta_{i\,j\,t}\mu_{t\,k} = \sum_{r=1}^{n} \sum_{s=1}^{n} \mu_{i\,r}\mu_{j\,s}\gamma_{r\,s\,k}$ for $1 \leq i, j, k \leq n$. The equality of $(X \cdot \psi(b_i))_{k\,j}$ and $(\phi(b_i) \cdot X)_{k\,j}$ for $1 \leq i, j, k \leq n$ follows directly from this.

We conclude from Proposition 2.2.8 that we can convert between regular matrix representations quite efficiently — in particular, at the cost of computing and inverting the matrix X, and then performing a small number of matrix multiplications.

We also consider the cost of computing a basis and structure constants for A from a more general description of A. Suppose we are given a set of matrices $a_1, a_2, \ldots, a_k \in M_{n\times n}(F)$, and that $A \subseteq M_{n\times n}(F)$ is the smallest associative algebra containing these matrices (where addition and multiplication in A are matrix addition and multiplication).

For $i \geq 0$, we define the subspace A_i of $M_{n\times n}(F)$ by

(i) $A_0 = \{\alpha I_n : \alpha \in F\}$;

(ii) A_i is the vector space spanned by the matrices $\prod_{h=1}^{i} a_{l_h}$, for $l_1, l_2, \ldots, l_i$ elements of $\{0, 1, \ldots, k\}$ and with $a_0 = I_n$. That is, A_i is spanned by the set of products of at most i of the matrices $a_1, a_2, \ldots, a_k$.

Lemma 2.2.9 shows how these subspaces can be used to find a basis for the algebra A.

Lemma 2.2.9. Let $a_1, a_2, \ldots, a_k \in M_{n\times n}(F)$ and let $A, A_0, A_1, \ldots$ be as above.

(i) $a_1, a_2, \ldots, a_k \in A_i \subseteq A$ for all $i > 0$.
(ii) $A_i \subseteq A_{i+1}$ for all $i \geq 0$.
(iii) If $A_i = A_{i+1}$ for some $i > 0$ then $A_i = A_j = A$ for all $j \geq i$.
(iv) For all $i > 0$, if $\{ b_1, b_2, \ldots, b_l \}$ is a basis for A_i over F then A_{2i} is spanned by the matrices $b_r \cdot b_s$ for $1 \leq r, s \leq l$.
(v) $A_{n^2-1} = A$.

Proof. Part (i) follows by the definition of A_i and the fact that the algebra A contains the matrices $I_n, a_1, a_2, \ldots, a_k$ and is closed under addition and multiplication.

Parts (ii) and (iv) are clearly consequences of the definition of A_i.

Suppose $A_i = A_{i+1}$ for some $i \geq 0$. It is easily shown that $A_{i+1} = A_{i+2}$; it follows by induction on j that $A_i = A_j$ for all $j \geq i$. Now let $a, b \in A_i$; then $ab \in A_{2i} = A_i$. Hence A_i includes $I_n, a_1, a_2, \ldots, a_k$ and is closed under addition and multiplication; hence $A_i \supseteq A$. Since $A_i \subseteq A$ (by (i)), we have established part (iii).

Finally, we note that the dimension of A_{i+1} is greater than that of A_i if $A_i \neq A_{i+1}$. Since A_0 has dimension 1 over F, it follows by (i)–(iii) that either $A_i = A$ or A_i has dimension at least $i + 1$ over F for all $i \geq 0$. Now $A_i \subseteq A \subseteq M_{n\times n}(F)$, and $M_{n\times n}(F)$ has dimension n^2 over F; part (v) follows. ∎

Theorem 2.2.10. Let $a_1, a_2, \ldots, a_k \in M_{n\times n}(F)$. A basis and set of structure constants for the algebra A generated by these matrices can be computed using a polynomial number of field operations, or by using Arithmetic-Boolean circuits of polynomial size and depth $O(\log^3(nk))$.

Proof. By Lemma 2.2.9 (v), $A = A_{n^2-1}$ for subspaces $A_0, A_1, \ldots$ as defined above. We compute a basis for $A_1 = A_{2^0}$ by finding a maximal linearly independent subset of $\{ I_n, a_1, a_2, \ldots, a_k \}$. This selection can be performed by checking whether each element of this set is a linear combination of the preceding elements — by solving at most $k+1$ systems, each of at most n^2 linear equations in at most k variables. This computation can be performed using time polynomial in nk, or using arithmetic-Boolean circuits over F of size polynomial in nk and with depth $O(\log^2(nk))$ (see Section 1.3 for details).

Suppose now that we have computed a basis $\{ b_1, b_2, \ldots, b_l \}$ for the vector space A_{2^i}; by part (iv) of Lemma 2.2.9 we can compute a basis for $A_{2^{i+1}}$ by choosing a maximal linearly independent subset of the matrices $\{ b_r \cdot b_s : 1 \leq r, s \leq l \}$. Let $h = \lceil \log_2(n^2 - 1) \rceil$. By Lemma 2.2.9 (iii) and (v), $A = A_{2^i}$ for $i \geq h$. We compute a basis for A by computing bases for $A_{2^0}, A_{2^1}, \ldots, A_{2^h}$.

Selection of a basis of $A_{2^{i+1}}$ given one for A_{2^i} involves multiplication of at most n^4 pairs of $n \times n$ matrices, followed by the selection of a maximal linearly independent subset of the products. This selection can be performed by solving at most n^4 systems, each of at most n^2 linear equations in at most n^4 variables over F. This computation can be performed using a polynomial number of field operations, or using Arithmetic-Boolean circuits over F of polynomial size and depth $O(\log^2 n)$. Since $h \in O(\log n)$, it is clear that a basis for A can be computed at the stated cost.

Using this basis, structure constants can be computed for A by solving at most n^4 systems of linear equations, each having a coefficient matrix of order n^2. Again, this computation can be performed at the stated cost. ■

2.2.3. Extension of Scalars

When discussing Boolean computations, we consider finite-dimensional associative algebras over $\mathbb{Q}$, algebraic number fields, and finite fields. These fields have succinct (Boolean) descriptions — and elements of these fields have useful representations (which are discussed in Section 1). Thus we can specify these fields, and finite-dimensional algebras over them, as inputs for Boolean algorithms.

We will also consider computations for finite-dimensional associative algebras over $\mathbb{R}$, $\mathbb{C}$, and algebraic closures of finite fields. In order to make representation of these algebras feasible, we restrict the set of algebras to be considered in the manner described below.

Definition 2.2.11. Suppose A and B are vector spaces over a field F, and consider the set $S(A, B)$ of (finite) formal sums $\sum_{i=1}^{n}(a_i, b_i)$ of pairs of elements $a_i \in B$ and $b_i \in B$, $1 \leq i \leq n$, with addition associative and commutative. Clearly, $S(A, B)$ is an (additive) Abelian group.

Let H be the subgroup of $S(A, B)$ generated by the formal sums

(i) $(a_1 + a_2, b) - (a_1, b) - (a_2, b)$

(ii) $(a, b_1 + b_2) - (a, b_1) - (a, b_2)$

(iii) $(a, \alpha b) - (a\alpha, b)$

for all $a, a_1, a_2 \in A$, $b, b_1, b_2 \in B$, and $\alpha \in F$. The *tensor product* of A and B, $A \otimes_F B$, is the factor group $S(A, B)/H$. For $a \in A$ and $b \in B$, we denote by $a \otimes_F b$ the element $(a, b) + H$ of $A \otimes_F B$. $A \otimes_F B$ forms a vector space over F, where we perform multiplication by a scalar using the rule

$$\alpha \cdot (\sum_{i=1}^{n}(a_i \otimes_F b_i)) = \sum_{i=1}^{n}((\alpha a_i) \otimes_F b_i) = \sum_{i=1}^{n}(a_i \otimes_F (\alpha b_i)).$$

We state without proof some useful facts about tensor products. (See Section 12 of Curtis and Reiner [31] for more details.)

Proposition 2.2.12. If A is a vector space with dimension n and with a basis $a_1, a_2, \ldots, a_n$ over F, and B is a vector space with dimension m and with a basis $b_1, b_2, \ldots, b_m$ over F, then $A \otimes_F B$ is a vector space with dimension nm and a basis $a_i \otimes_F b_j$ (for $1 \leq i \leq n$, $1 \leq j \leq m$) over F.

Suppose now that A and B are both algebras over F. We perform "multiplication" in $A \otimes_F B$ by multiplying pairs $a_1 \otimes_F b_1$ and $a_2 \otimes_F b_2$ componentwise (for $a_1, a_2 \in A$, $b_1, b_2 \in B$) and using the distributive law for multiplication over addition to obtain a (well defined) product of an arbitrary pair of elements of $A \otimes_F B$:

$$\begin{aligned} \left(\sum_{i=1}^{r}(a_{1\,i} \otimes_F b_{1\,i})\right) \cdot \left(\sum_{j=1}^{s}(a_{2\,j} \otimes_F b_{2\,j})\right) &= \sum_{i=1}^{r}\sum_{j=1}^{s}((a_{1\,i} \otimes_F b_{1\,i}) \cdot (a_{2\,j} \otimes_F b_{2\,j})) \\ &= \sum_{i=1}^{r}\sum_{j=1}^{s}((a_{1\,i}a_{2\,j}) \otimes_F (b_{1\,i}b_{2\,j})). \end{aligned}$$

With this definition of multiplication, we can show that $A \otimes_F B$ is an associative algebra over F if A and B are. Again, $A \otimes_F B$ is finite-dimensional if both A and B are.

We use the tensor product construction for a different reason — to obtain an algebra A_E over a field extension $E \supseteq F$ from an algebra A over a field F, by "extension of scalars".

Proposition 2.2.13. If A is an associative algebra over F, and E is an extension field of F, then $A \otimes_F E$ is an associative algebra over E with multiplication in $A \otimes_F E$ as defined above (viewing E as a vector space over F), and with multiplication by a scalar (in E) defined as follows.

$$\alpha \cdot \left(\sum_{i=1}^{r}(c_i \otimes_F e_i)\right) = \left(\sum_{i=1}^{r}(c_i \otimes_F (\alpha e_i))\right),$$

for $c_1, c_2, \ldots, c_r \in A$, $e_1, e_2, \ldots, e_r, \alpha \in E$. If A has dimension n and basis $a_1, a_2, \ldots, a_n$ over F, then $A \otimes_F E$ has dimension n and a basis

$$(a_1 \otimes_F 1_E), (a_2 \otimes_F 1_E), \ldots, (a_n \otimes_F 1_E)$$

over E (for 1_E the multiplicative identity in E).

Furthermore, if the set $\{\, \gamma_{i\,j\,k} : 1 \leq i, j, k \leq n \,\}$ is a set of structure constants for A with respect to the basis $a_1, a_2, \ldots, a_n$, then $\{\, \gamma_{i\,j\,k} : 1 \leq i, j, k \leq n \,\}$ is also a set of structure constants for the algebra $A \otimes_F E$ with respect to the basis $(a_1 \otimes_F 1_E)$, $(a_2 \otimes_F 1_E)$, $\ldots$, $(a_n \otimes_F 1_E)$.

We abbreviate $A \otimes_F E$ to A_E when the ground field F is known.

When considering algebras over $\mathbb{R}$ or $\mathbb{C}$, we only consider algebras $A \otimes_F \mathbb{R}$ or $A \otimes_F \mathbb{C}$, where F is a number field and A is a finite-dimensional associative algebra over F. The only algebras over algebraic closures of finite fields we consider are of the form $A \otimes_F E$, where F is a finite field, E an algebraic closure of F, and A a finite-dimensional associative algebra over F. By Proposition 2.2.13, these have succinct (Boolean) descriptions — namely, those given by a set of structure constants in a finite algebraic extension of a prime field.

As the example below indicates, it is *not* generally sufficient to consider the algebra A when decomposing $A \otimes_F E$.

Example 2.2.14. Let $F = \mathbb{Q}$, and let A be the associative algebra of dimension 4 over F with basis $\{ 1, i, j, k \}$ and regular matrix representation (and structure constants) shown in Example 2.2.6. If $E = \mathbb{R}$, then A_E is an associative algebra of dimension 4 over $\mathbb{R}$ — the algebra of real quaternions. If $E = \mathbb{C}$, then A_E is again an associative algebra of dimension 4 — over $\mathbb{C}$. We will show later that (for $E = \mathbb{C}$) A_E is isomorphic to the matrix ring $M_{2\times 2}(\mathbb{C})$.

We obtain two sets of problems for decompositions of algebras A over fields F corresponding to Theorems 2.1.12, 2.1.23, and 2.1.25. We consider the computation of decompositions of A over the original field F, as well as decompositions of $A \otimes_F E$, for $E \supset F$ (We will consider cases E real and algebraically closed). These problems are discussed further in Sections 2.3–2.5.

2.3. Computation of the Radical

We consider algorithms for computation of the radical of a finite-dimensional associative algebra A over a field F. As stated in Section 2.1, the radical of A is also a subspace of the vector space A over F — so it can be represented by a basis over F. We will also produce a basis over F for the factor algebra $A/\mathrm{rad}\,(A)$ when isolating the radical.

Problem **Isolation of the Radical**.

Input.
- Integers n, $m > 0$.
- Matrices $a_1, a_2, \ldots, a_n \in M_{m\times m}(F)$, which form the basis for a finite-dimensional associative algebra $A \subseteq M_{m\times m}(F)$ of dimension n over F.

Output.
- Integer $r \geq 0$, the dimension of $\mathrm{rad}(A)$ over F.
- Elements $\mu_{i\,j}$ of F, for $1 \leq i, j \leq n$, which define elements $b_1, b_2, \ldots, b_n \in A$, with $b_i = \sum_{j=1}^{n} \mu_{i\,j}\, a_j$ for $1 \leq i \leq n$, so that
  (1) $b_1, b_2, \ldots, b_r$ is a basis for $\mathrm{rad}(A)$ over F;
  (2) $b_1, b_2, \ldots, b_n$ is a basis for A over F.
- Matrices $c_1, c_2, \ldots, c_{n-r} \in M_{(n-r)\times(n-r)}(F)$ forming the basis for a semi-simple associative algebra over F isomorphic to the factor algebra $A/\mathrm{rad}(A)$.

We will see later that the elements $b_{r+1}, b_{r+2}, \ldots, b_n$ can be used to obtain a basis for the factor algebra $A/\mathrm{rad}(A)$. We will find a matrix representation for $A/\mathrm{rad}(A)$ by generating the regular representation with respect to this basis.

Friedl and Rónyai [43] show that this problem can be solved efficiently when F is a finite field or a finite algebraic extension of $\mathbb{Q}$, when using Boolean computations. We will review their methods. In the process, we show that the computations can be performed efficiently when using an arithmetic (rather than Boolean) model, and that the computations can be performed efficiently in parallel. We will also verify that Friedl and Rónyai's algorithms can be applied directly to compute the radical of an algebra $A \otimes_F E$ over E, for A a finite-dimensional algebra over a number field F and for $E = \mathbb{R}$ or $E = \mathbb{C}$. Finally, we will reduce the problem of computing the squarefree decomposition of a polynomial over a field F to the computation of the radical of a finite-dimensional algebra over F, in order to conclude that we *cannot* compute the radical of finite-dimensional associative algebras over arbitrary fields.

2.3.1. Computations over Fields of Characteristic Zero

As Friedl and Rónyai note, the problem is relatively simple for fields of characteristic zero. The radical of a finite-dimensional associative algebra over such a field can be characterised as the set of solutions of a homogeneous system of linear equations. We begin by developing this characterisation.

Let $a \in A$, and suppose the matrix $\phi(a) \in M_{m\times m}(F)$ has characteristic polynomial

$$\chi(a) = t^m + \lambda_{m-1}t^{m-1} + \lambda_{m-2}x^{m-2} + \cdots \lambda_1 t + \lambda_0 = \det\,(tI_m - \phi(a)),$$

for coefficients $\lambda_{m-1}, \lambda_{m-2}, \ldots, \lambda_1, \lambda_0 \in F$. Suppose also that the matrix $\phi(a)$ has characteristic values $\psi_1, \psi_2, \ldots, \psi_m$ in some algebraic closure of F. Then

$$\chi(a) = \prod_{i=1}^{m}(t - \psi_i).$$

The coefficients of the characteristic polynomial are the values of the elementary symmetric polynomials at the negatives of these characteristic values:

$$\lambda_i = \sum_{\substack{I\subseteq\{\,1,\,2,\,\ldots,\,m\,\}\\ |I|=i}} \prod_{h\in I}(-\psi_h).$$

In particular, the *trace* of $\phi(a)$, $\sum_{h=1}^{m}\psi_h$, is $-\lambda_{m-1}$ (as well as the sum of the diagonal entries of $\phi(a)$), while the *norm* of $\phi(a)$, $\prod_{h=1}^{m}\psi_h$, is $(-1)^m\lambda_0 = \det\,\phi(a)$.

Suppose now that a is nilpotent in A — so that $a^i = 0$ for some $i \geq 0$. This implies that the characteristic values ψ_j $(1 \leq j \leq m)$ of a are all zero (since the matrix a^i has characteristic values ψ_j^i for $1 \leq j \leq m$). Thus we have

$$\chi(a) = \prod_{i=1}^{m}(t - 0) = t^m$$

if a is nilpotent. In fact it is clear that the converse also holds.

Proposition 2.3.1. An element a of A is nilpotent if and only if a has characteristic polynomial t^m. Hence a is nilpotent if and only if $a^m = 0$.

Thus we have an efficient procedure to test whether an element a of A is nilpotent. Dickson [33] gives a simple criterion for membership in the radical of A.

Theorem 2.3.2. (Dickson.) Let A be a finite-dimensional associative algebra over a field F of characteristic zero; then an element a of A is in the radical of A if and only if the trace of ax is 0 for all $x \in A$.

Proof. Suppose $a \in A$ is in the radical of A. Then a is strongly nilpotent, and ax is nilpotent for any $x \in A$. Thus the characteristic values, and hence the trace, of ax are all zero.

Suppose now that $a \in A$ such that the trace of ax is zero for all $x \in A$. We wish to prove that ax is nilpotent for all x. Fix x; then it is sufficient to show that the characteristic values $\psi_1, \psi_2, \ldots, \psi_m$ of ax are all zero. We know that $(ax)^i$ has trace zero for all $i > 0$, since $(ax)^i = az$, for $z = x(ax)^{i-1} \in A$. Since the characteristic values of $(ax)^i$ are the i^{th} powers of the characteristic values of ax, we see that

$$\sum_{j=1}^{m} \psi_j^i = 0 \qquad \text{for all } i > 0.$$

Thus all of the *power sum symmetric functions* have value 0 when evaluated at these characteristic values. The *elementary symmetric functions* are all $\mathbb{Q}$-linear combinations of the power sum symmetric functions (see Stanley [113] for details). Hence these also have value 0 at the characteristic values of ax. Thus the coefficients $\lambda_{m-1}, \lambda_{m-2}, \ldots, \lambda_1, \lambda_0$ of the characteristic polynomial of ax are 0. It follows that ax has characteristic polynomial t^m, and characteristic values all zero, as required. ∎

Before continuing with our discussion of the case F has characteristic 0, we note that the proof of Theorem 2.3.2 fails for the case F has positive characteristic p, because the elementary symmetric functions are *not* generally $\mathbb{F}_p$-linear combinations of the power sum symmetric functions; again, see Stanley [113] for details.

Friedl and Rónyai observe that Theorem 2.3.2 gives an efficient test for membership in the radical, and for computation of the basis of the radical, for algebras over fields of characteristic zero.

Corollary 2.3.3. (Friedl and Rónyai). Let $A \subseteq M_{m\times m}(F)$ be a finite-dimensional associative algebra over a field F of characteristic zero, with basis $\{ a_1, a_2, \ldots, a_n \}$ over F. Then an element a of A is a member of the radical of A if and only if the trace of aa_i is zero for all i, $1 \le i \le n$.

Proof. It is sufficient to note that the trace is a linear function; hence the function

$$\text{Trace}(a \cdot (\lambda_1 a_1 + \lambda_2 a_2 + \cdots + \lambda_n a_n))$$

is a linear function of the indeterminates $\lambda_1, \lambda_2, \ldots, \lambda_n$. ∎

We compute a basis for the radical of a finite-dimensional associative algebra A with basis $\{ a_1, a_2, \ldots, a_n \}$ over a field F of characteristic zero by using Corollary 2.3.3 to obtain a system of linear equations with solution set

$$\{ (\lambda_1, \lambda_2, \ldots, \lambda_n) \in F^n \, : \, \lambda_1 a_1 + \lambda_2 a_2 + \cdots + \lambda_n a_n \in \mathrm{rad}(A) \},$$

and solving this system, as shown in the following algorithm.

Algorithm **Isolation of the Radical — Characteristic Zero**

Input.
- Integers n, $m > 0$.
- Matrices $a_1, a_2, \ldots, a_n \in M_{m\times m}(F)$, which form the basis for a finite-dimensional associative algebra $A \subseteq M_{m\times m}(F)$ of dimension n over a field F of characteristic zero.

Output.
- Integer $r \geq 0$, the dimension of $\mathrm{rad}(A)$ over F.
- Elements $\mu_{i\,j}$ of F, for $1 \leq i, j \leq n$, which define elements $b_1, b_2, \ldots, b_n \in A$, with $b_i = \sum_{j=1}^{n} \mu_{i\,j}\, a_j$ for $1 \leq i \leq n$, so that
  (1) $b_1, b_2, \ldots, b_r$ is a basis for $\mathrm{rad}(A)$ over F;
  (2) $b_1, b_2, \ldots, b_n$ is a basis for A over F.
- Matrices $c_1, c_2, \ldots, c_{n-r} \in M_{(n-r)\times(n-r)}(F)$ forming a basis for a semi-simple associative algebra over F isomorphic to the factor algebra $A/\mathrm{rad}(A)$.

(1) Compute the dimension r and a basis $\{ \hat{b}_1, \hat{b}_2, \ldots, \hat{b}_r \}$ over F for the space of solutions in F^n of the system of n linear equations
$\mathrm{Trace}((\lambda_1 a_1 + \lambda_2 a_2 + \cdots + \lambda_n a_n) a_i) = 0$
for $1 \leq i \leq n$, and for indeterminates $\lambda_1, \lambda_2, \ldots, \lambda_n$.

(2) Extend this basis to obtain a basis $\{ \hat{b}_1, \hat{b}_2, \ldots, \hat{b}_n \}$ for F^n over F, by adding each element e_i $((0, \ldots, 0, 1, 0, \ldots, 0)$, with 1 as the i^{th} coordinate) which is not an F-linear combination of the elements $\hat{b}_1, \hat{b}_2, \ldots, \hat{b}_r, e_1, e_2, \ldots, e_{i-1}$, for $1 \leq i \leq n$. Set $\mu_{i\,j}$ to be the j^{th} coordinate of the vector $\hat{b}_i$, for $1 \leq i, j \leq n$.

(3) Let $b_i = \mu_{i\,1} a_1 + \mu_{i\,2} a_2 + \cdots + \mu_{i\,n} a_n$, for $1 \leq i \leq n$. The algebra $A/\mathrm{rad}(A)$ has a basis $b_{r+1} + \mathrm{rad}(A)$, $b_{r+2} + \mathrm{rad}(A)$, $\cdots$, $b_n + \mathrm{rad}(A)$. Compute a set of structure constants for $A/\mathrm{rad}(A)$ with respect to this basis, and set $c_i = \phi(b_{r+i} + \mathrm{rad}(A))$, for $1 \leq i \leq n - r$, and for ϕ the regular representation for $A/\mathrm{rad}(A)$ with respect to this basis.

Theorem 2.3.4. Let $A \subseteq M_{m\times m}(F)$ be a finite-dimensional associative algebra of dimension n over a field F of characteristic zero, for m, $n > 0$.

(i) A basis for the radical of A, and the remaining output of the problem "Isolation of the Radical", can be computed from a basis for A over F, using a polynomial number of field operations, or in parallel using arithmetic-Boolean circuits of depth $O(\log^2(nm))$ and of polynomial size.

(ii) If F is a finite algebraic extension of $\mathbb{Q}$, and A is as above, then the output of the problem "Isolation of the Radical" can be computed using a polynomial number of Boolean operations, or in parallel using Boolean circuits of depth $O(\log^2 N)$ and of polynomial size (for input size N).

Proof. We use the algorithm "Isolation of the Radical — Characteristic Zero" to solve this problem over fields of characteristic zero, by solving systems of linear equations of size polynomial in the number of inputs. Hence the cost of computing the output is dominated by the cost of solving systems of linear equations of polynomial size. The bounds stated in part (i) of the theorem follow immediately (see Section 1.3 for details).

As shown in Section 1.3, computation of the solutions of systems of linear equations over finite algebraic extensions of $\mathbb{Q}$ can be reduced to the computation of determinants of matrices (of polynomial size) with entries in $\mathbb{Q}$. Hence the bounds stated in part (ii) of the theorem follow from well known results about the computation of determinants (which are also discussed in Section 1.3). ■

As shown in Section 2.2, a matrix representation (with $m = n$) can be computed from a set of structure constants for A — so that the above values can also be computed at the stated cost from a set of structure constants.

Example 2.3.5. Consider the algebra A of 2×2 upper triangular matrices over $F = \mathbb{Q}$. Suppose we are given the basis

$$a_1 = \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix}, \qquad a_2 = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \qquad a_3 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}.$$

To compute a basis for the radical of A, we form the coefficient matrix for the system of equations

$$\text{Trace}((\lambda_1 a_1 + \lambda_2 a_2 + \lambda_3 a_3) \cdot a_i) = 0, \qquad \text{for } 1 \le i \le 3.$$

Computing representations, we have

$$\begin{aligned}
(\lambda_1 a_1 + \lambda_2 a_2 + \lambda_3 a_3) &= \begin{bmatrix} \lambda_1 & \lambda_1 + \lambda_2 \\ 0 & \lambda_3 \end{bmatrix}, \\
((\lambda_1 a_1 + \lambda_2 a_2 + \lambda_3 a_3) \cdot a_1) &= \begin{bmatrix} \lambda_1 & \lambda_1 \\ 0 & 0 \end{bmatrix}, \\
((\lambda_1 a_1 + \lambda_2 a_2 + \lambda_3 a_3) \cdot a_2) &= \begin{bmatrix} 0 & \lambda_1 \\ 0 & 0 \end{bmatrix}, \\
((\lambda_1 a_1 + \lambda_2 a_2 + \lambda_3 a_3) \cdot a_3) &= \begin{bmatrix} 0 & \lambda_1 + \lambda_2 \\ 0 & \lambda_3 \end{bmatrix}.
\end{aligned}$$

Thus we obtain the system

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}.$$

Solving, we find that $\left\{ \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \right\}$ is a basis for the set of solutions — so that the radical has dimension 1, and basis $\{ a_2 \}$ over $\mathbb{Q}$.

We extend this to obtain a basis for A by considering each element of the original basis $\{ a_1, a_2, \ldots, a_n \}$, adding each element which is not a linear combination of the elements already added to the new basis. Proceeding in this way, we obtain the basis $\{b_1, b_2, b_3 \}$ with $b_1 = a_2$, $b_2 = a_1$, and $b_3 = a_3$. Hence we obtain the elements

$$\begin{array}{lll}
\mu_{1\,1} = 0 & \mu_{1\,2} = 1 & \mu_{1\,3} = 0 \\
\mu_{2\,1} = 1 & \mu_{2\,2} = 0 & \mu_{2\,3} = 0 \\
\mu_{3\,1} = 0 & \mu_{3\,2} = 0 & \mu_{3\,3} = 1
\end{array}$$

It remains only for us to find a basis and matrix representation for the factor algebra $A/\mathrm{rad}(A)$. Clearly we can take

$$\hat{c}_1 = b_2 + \mathrm{rad}(A), \qquad \hat{c}_2 = b_3 + \mathrm{rad}(A)$$

as our new basis. We compute structure constants for the basis $\{ \hat{c}_1, \hat{c}_2 \}$ by checking the products $b_i \cdot b_j$ for $i, j \in \{ 2, 3 \}$:

$$\begin{aligned}
b_2 \cdot b_2 &= \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix} = 1 \cdot b_2 + 0 \cdot b_3 + 0 \cdot b_1, \\
b_2 \cdot b_3 &= \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} = 0 \cdot b_2 + 0 \cdot b_3 + 1 \cdot b_1, \\
b_3 \cdot b_2 &= \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} = 0 \cdot b_2 + 0 \cdot b_3 + 0 \cdot b_1, \\
b_3 \cdot b_3 &= \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} = 0 \cdot b_2 + 1 \cdot b_3 + 0 \cdot b_1.
\end{aligned}$$

Since $b_1 \in \mathrm{rad}(A)$, we ignore the coefficients in this element when forming structure constants for $A/\mathrm{rad}(A)$. Hence we obtain the basis $\{ c_1, c_2 \}$ for $A/\mathrm{rad}(A)$, with

$$c_1 = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, \qquad c_2 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}.$$

Example 2.3.6. Consider the algebra A of Example 2.3.5. Suppose we are given as input a set of structure constants for the basis $\{ a_1, a_2, a_3 \}$ of that example. That is, we are given a set of 27 structure constants, which yield the regular representation ϕ, with

$$\phi(a_1) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}, \quad \phi(a_2) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}, \quad \phi(a_3) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

To compute a basis for the radical of A, we form the coefficient matrix for the system of equations

$$\mathrm{Trace}((\lambda_1 a_1 + \lambda_2 a_2 + \lambda_3 a_3) \cdot a_i) = 0, \qquad \text{for } 1 \le i \le 3.$$

Computing representations, we have

$$\begin{aligned}
\phi(\lambda_1 a_1 + \lambda_2 a_2 + \lambda_3 a_3) &= \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_1 & \lambda_1 + \lambda_2 \\ 0 & 0 & \lambda_3 \end{bmatrix}, \\
\phi((\lambda_1 a_1 + \lambda_2 a_2 + \lambda_3 a_3) \cdot a_1) &= \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_1 & \lambda_1 \\ 0 & 0 & 0 \end{bmatrix}, \\
\phi((\lambda_1 a_1 + \lambda_2 a_2 + \lambda_3 a_3) \cdot a_2) &= \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & \lambda_1 \\ 0 & 0 & 0 \end{bmatrix}, \\
\phi((\lambda_1 a_1 + \lambda_2 a_2 + \lambda_3 a_3) \cdot a_3) &= \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & \lambda_1 + \lambda_2 \\ 0 & 0 & \lambda_3 \end{bmatrix}.
\end{aligned}$$

Thus we obtain the system

$$\begin{bmatrix} 2 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}.$$

Note that this is not the system obtained in the previous example. Solving, we (again) find that $\left\{ \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \right\}$ is a basis for the set of solutions — so that the radical has dimension 1, and basis $\{ a_2 \}$ over $\mathbb{Q}$.

We extend this to obtain a basis for A as in Example 2.3.5. Again, we obtain the basis $\{ b_1, b_2, b_3 \}$ with $b_1 = a_2$, $b_2 = a_1$, and $b_3 = a_3$. Hence we obtain the elements

$$\begin{array}{lll} \mu_{1\,1} = 0 & \mu_{1\,2} = 1 & \mu_{1\,3} = 0 \\ \mu_{2\,1} = 1 & \mu_{2\,2} = 0 & \mu_{2\,3} = 0 \\ \mu_{3\,1} = 0 & \mu_{3\,2} = 0 & \mu_{3\,3} = 1 \end{array}$$

and the matrix

$$X = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Finally, we invert this matrix to obtain

$$X^{-1} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} = X.$$

We use these matrices as discussed in Section 2.2.2 to obtain the regular representation ψ for A with respect to the new basis $\{ b_1, b_2, b_3 \}$.

$$\begin{aligned} \psi(b_1) = X^{-1}\phi(b_1)X &= \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}; \\ \psi(b_2) = X^{-1}\phi(b_2)X &= \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}; \\ \psi(b_3) = X^{-1}\phi(b_3)X &= \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \end{aligned}$$

Now since $\operatorname{rad}(A)$, with basis $\{ b_1, b_2, \ldots, b_r \}$, is a two-sided ideal in A, we see that the regular representation with respect to basis $\{ b_1, b_2, \ldots, b_n \}$ is block upper triangular, with an $r \times r$ upper block and an $(n - r) \times (n - r)$ lower block. For $1 \leq i \leq r$, the lower block of $\psi(b_i)$ is zero. We obtain a basis (and a regular representation with respect to this basis) for $A/\operatorname{rad}(A)$ from the lower $(n-r)\times(n-r)$

blocks of the matrices $\psi(b_{r+1})$, $\psi(b_{r+2})$, $\ldots$, $\psi(b_n)$. In our example, we obtain the matrices

$$c_1 = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \qquad \text{and} \qquad c_2 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix},$$

the same as those obtained in Example 2.3.5.

As indicated in the above examples, we can work directly with the $m \times m$ matrices given to us as input — or we can generate a regular representation ($n \times n$ matrices) and work with those. Clearly the former approach is (slightly) more efficient for $m < n$, while the latter could save time for $m > n$.

2.3.2. Computations over Finite Fields

We now consider algebras over fields of positive characteristic, and, in particular, over finite fields. We first note that the algorithm stated for fields of characteristic zero is not correct when applied to algebras over fields of positive characteristic.

Example 2.3.7. Consider the algebra A of upper triangular 2×2 matrices over $\mathbb{F}_2$, and suppose we are given the structure constants for the basis

$$a_1 = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, \qquad a_2 = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \qquad a_3 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$$

(as in Example 2.3.6). As showed in Example 2.1.10, the radical of A contains the ideal generated by a_2. We note that the radical is spanned by a_2: for if

$$a = \begin{bmatrix} \alpha & \beta \\ 0 & \gamma \end{bmatrix} \in \mathrm{rad}(A),$$

then $\alpha = 0$, since $aa_1 = \alpha a_1$ is nilpotent (while a_1 is not), and $\gamma = 0$, since $a_3 a = \gamma a_3$ is nilpotent (while a_3 is not). However, the algorithm for isolation of the radical of algebras over fields of characteristic zero computes as the radical the set $\lambda_1 a_1 + \lambda_2 a_2 + \lambda_3 a_3$ for λ_1, λ_2, $\lambda_3 \in \mathbb{F}_2$ such that

$$\begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix};$$

in particular, this test incorrectly identifies a_1 as an element of the radical of A.

Friedl and Rónyai generalise Dickson's criterion, to obtain an algorithm for computation of the radical of a finite-dimensional associative algebra A over a finite field

$\mathbb{F}_{p^l}$ of characteristic $p > 0$. They reduce the problem to "Isolation of the Radical" over prime fields, by deriving from A a finite-dimensional associative algebra $\hat{A} \subseteq M_{ml \times ml}(\mathbb{F}_p)$ of dimension nl over $\mathbb{F}_p$, whose radical can be used to obtain $\mathrm{rad}(A)$. They then present an algorithm for "Isolation of the Radical" of algebras over $\mathbb{F}_p$. We generalise their algorithm for "Isolation of the Radical" to arbitrary finite fields, in hopes of improving the efficiency of the algorithm over $\mathbb{F}_{p^l}$, for large l.

Now let $A \subset M_{m\times m}(F)$ be a finite-dimensional associative algebra of dimension n over $\mathbb{F}_{p^l}$. Let $k = \lfloor \log_p m \rfloor \in \mathbb{N}$ — so that $p^k \le m < p^{k+1}$. As is the case for Friedl and Rónyai's method, we obtain a basis for the radical of A by finding ideals $I_{-1}, I_0, I_1, \ldots, I_k$ of A and functions $g_i : I_{i-1} \to F_{p^l}$ for $0 \le i < k$, so that

(i) $I_{-1} = A$ and $I_k = \mathrm{rad}(A)$;

(ii) g_i is an $\mathbb{F}_p$-linear function on I_{i-1};

(iii) $I_i = \{\, a \in I_{i-1} \,:\, g_i(ab) = 0 \text{ for all } b \in A \,\}$.

Hence we will compute a basis for the radical of A over $\mathbb{F}_{p^l}$ using arithmetic in $\mathbb{F}_p$, to compute bases for the ideals $I_0, I_1, \ldots, I_k$ over $\mathbb{F}_p$, by solving systems of $\mathbb{F}_p$-linear equations defined using the function g_i. We now define this function.

Recall that the field $\mathbb{F}_{p^l}$ is isomorphic to $\mathbb{F}_p[t]/(f)$, for some monic irreducible polynomial $f \in \mathbb{F}_p[t]$ with degree l. Let $\hat{f} \in \mathbb{Z}[t]$ be the monic polynomial with degree l and with coefficients between 0 and $p-1$ whose coefficients mod p are those of f; that is, $f = (\hat{f} \bmod p)$. Since f is irreducible in $\mathbb{F}_p[t]$ it is clear that $\hat{f}$ is irreducible in $\mathbb{Z}[t]$, and hence in $\mathbb{Q}[t]$. Since $\hat{f}$ is monic, we see that $\mathbb{Z}[t]/(\hat{f})$ is an integral domain, and a subring of the ring of algebraic integers in the number field $\mathbb{Q}[t]/(\hat{f})$. We have a natural (ring) homomorphism $\rho : \mathbb{Z}[t] \to \mathbb{F}_p[t]$ taking integer polynomials to their residues mod p; since the image (under ρ) of the ideal $(\hat{f}) \subseteq \mathbb{Z}[t]$ is the ideal $(f) \subseteq \mathbb{F}_p[t]$, we also have an induced (ring) homomorphism

$$\rho : \mathbb{Z}[t]/(\hat{f}) \to \mathbb{F}_p[t]/(f) \cong \mathbb{F}_{p^l},$$

with

$$\rho : (h \bmod (\hat{f})) \mapsto (\rho(h) \bmod (f))$$

for $h \in \mathbb{Z}[t]$. We define the map g_i (on the subspace I_{i-1} of A) by describing a map $\hat{g}_i$ from $m \times m$ matrices with entries in $\mathbb{Z}[t]/(\hat{f})$ to $\mathbb{Q}[t]/(\hat{f})$. We show that if α is an $m \times m$ matrix with entries in $\mathbb{Z}[t]/(\hat{f})$ such that $(\alpha \bmod p) \in I_{i-1}$, then $\hat{g}_i(\alpha) \in \mathbb{Z}[t]/(\hat{f})$. The map g_i will then be defined so that

$$g_i(\alpha \bmod p) = \hat{g}_i(\alpha) \bmod p$$

for such a matrix α. We continue by showing that the map g_i is well defined, and then showing that it is $\mathbb{F}_p$-linear on the ideal I_{i-1}, so that it can be used to generate a set I_i as in the above algorithm. We then show that I_i is a subspace of A over $\mathbb{F}_{p^l}$.

For $0 \leq i \leq k$, we define the map $\hat{g}_i : M_{m\times m}(\mathbb{Z}[t]/(\hat{f})) \rightarrow \mathbb{Q}[t]/(\hat{f})$ by

$$\hat{g}_i(\alpha) = \frac{\text{Trace }(\alpha^{p^i})}{p^i} \qquad \text{for } \alpha \in M_{m\times m}(\mathbb{Z}[t]/(\hat{f})).$$

We define the set $\hat{I}_i \subseteq M_{m\times m}(\mathbb{Z}[t]/(\hat{f}))$ inductively for $-1 \leq i \leq k$.

(i) $\hat{I}_{-1} = \{\, \alpha \in M_{m\times m}(\mathbb{Z}[t]/(\hat{f})) \,:\, (\alpha \bmod p) \in A \,\}$;

(ii) $\hat{I}_{i+1} = \{\, \alpha \in I_i \;:\; \hat{g}_{i+1}(\alpha\beta) \in \mathbb{Z}[t]/(\hat{f}) \qquad\qquad \text{for all } \beta \in I_{-1} \,\}$ for $-1 \leq i < k$.

Lemma 2.3.8. If α, $\beta \in M_{m\times m}(\mathbb{Z}[t]/(\hat{f}))$ and $\alpha \equiv \beta \pmod p$, then

$$\text{Trace}(\alpha^{p^i}) \equiv \text{Trace}(\beta^{p^i}) \pmod{p^{i+1}} \qquad \text{for all } i \geq 0.$$

Proof. Friedl and Rónyai prove this result for the case $l = 1$ — so that α and β are integer matrices (see Friedl and Rónyai [43], Lemma 5.1). We note that their proof is also correct for matrices with entries in the domain $\mathbb{Z}[t]/(\hat{f})$. ■

It follows immediately from this lemma that if α, $\beta \in M_{m\times m}(\mathbb{Z}[t]/(\hat{f}))$
and $\alpha \equiv \beta \pmod p$, then

$$\frac{1}{p^{i+1}}(\text{Trace}(\alpha^{p^i}) - \text{Trace}(\beta^{p^i})) \in \mathbb{Z}[t]/(\hat{f}).$$

Lemma 2.3.9. If $i \geq 0$ and $\alpha \in \hat{I}_{i-1}$ then for all $\beta \in \hat{I}_{-1}$,

$$\frac{\text{Trace}((\alpha\,\beta)^{p^i})}{p^i} \in \mathbb{Z}[t]/(\hat{f}).$$

Proof. If $i = 0$ then

$$\frac{\text{Trace}((\alpha\,\beta)^{p^i})}{p^i} = \text{Trace}(\alpha\,\beta),$$

and the result is obvious.

Suppose now that $i > 0$, $\alpha \in \hat{I}_{i-1}$, and $\beta \in \hat{I}_{-1}$. Since α, $\beta \in M_{m\times m}(\mathbb{Z}[t]/(\hat{f}))$, $p^i\hat{g}_i(\alpha\,\beta) = \text{Trace}((\alpha\,\beta)^{p^i}) \in \mathbb{Z}[t]/(\hat{f})$. Since $\alpha \in \hat{I}_{i-1}$, $\hat{g}_{i-1}(\alpha\,\gamma) \in \mathbb{Z}[t]/(\hat{f})$ and $\hat{g}_{i-1}(\alpha\,\gamma) \equiv 0 \pmod p$ for all $\gamma \in \hat{I}_{-1}$. In particular (using $\gamma = \beta(\alpha\,\beta)^{p-1}$),

$$\hat{g}_{i-1}((\alpha\,\beta)^p) \in \mathbb{Z}[t]/(\hat{f}) \qquad \text{and} \qquad \hat{g}_{i-1}((\alpha\,\beta)^p) \equiv 0 \pmod p.$$

That is,

$$\frac{\mathrm{Trace}((\alpha\,\beta)^{p^i})}{p^{i-1}} \in \mathbb{Z}[t]/(\hat{f}) \qquad \text{and} \qquad \frac{\mathrm{Trace}((\alpha\,\beta)^{p^i})}{p^{i-1}} \equiv 0 \pmod{p}.$$

Hence p^i divides $\mathrm{Trace}((\alpha\,\beta)^{p^i})$ in $\mathbb{Z}[t]/(\hat{f})$, as desired. ■

Lemmas 2.3.8 and 2.3.9 imply that if $-1 \leq i < k$ and $\alpha \in I_{i-1}$, then $\hat{g}_i(\alpha) \in \mathbb{Z}[t]/(\hat{f})$, and that if $\alpha, \beta \in I_{i-1}$ with $\alpha \equiv \beta \pmod{p}$, then $\hat{g}_i(\alpha) \equiv \hat{g}_i(\beta) \pmod{p}$. It is clear from these lemmas that the map $g_i : I_{i-1} \to \mathbb{F}_{p^l}$ and the set

$$I_i = \{\, a \in I_{i-1} \,:\, g_i(ab) = 0 \text{ for all } b \in A \,\}$$

are both well defined, for $i = 1, 2, \ldots, k$. (Note also that $I_i = \{\, \alpha \bmod p \,:\, \alpha \in \hat{I}_i \,\}$ for $-1 \leq i \leq k$).

It is clear from the definition of $\hat{g}_i$ that the set $\hat{I}_i$ is closed under multiplication, for $-1 \leq i \leq k$. Thus the following lemma can be applied to this set.

Lemma 2.3.10. Let H be a multiplicatively closed subset of $M_{m\times m}(\mathbb{Z}[t]/(\hat{f}))$, let $j \geq 0$, and suppose that $\mathrm{Trace}(\alpha^{p^i})$ is divisible by p^{i+1} for all $\alpha \in H$ and all i, $0 \leq i < j$. Then for every $\alpha, \beta \in H$,

$$\mathrm{Trace}((\alpha+\beta)^{p^j}) \equiv \mathrm{Trace}((\alpha)^{p^j}) + \mathrm{Trace}((\beta)^{p^j}) \pmod{p^{j+1}}.$$

Proof. Friedl and Rónyai prove this result for the case $l = 1$ (see Friedl and Rónyai [43], Lemma 5.2). That is, they prove the result for integer matrices α and β. We note that their proof generalises to the case $\alpha, \beta \in M_{m\times m}(\mathbb{Z}[t]/(\hat{f}))$. ■

Lemma 2.3.10 implies that if $a, b \in I_{i-1} \subseteq A$ then $g_i(a+b) = g_i(a)+g_i(b)$. It is also clear that if $a \in I_{i-1}$ and $\gamma \in \mathbb{F}_p$, then $g_i(\gamma a) = \gamma g_i(a)$, since $\gamma^{p^i} = \gamma$ for $\gamma \in \mathbb{F}_p$. Thus g_i is an $\mathbb{F}_p$-linear map on I_{i-1}. It is not generally true that g_i is $\mathbb{F}_{p^l}$-linear as well.

We note that the proofs of Lemma 5.3, Lemma 5.4, Theorem 5.5, and Theorem 5.6 of Friedl and Rónyai (for the case $l = 1$) are valid for the general case. We state the more general versions of these lemmas and theorems below.

Lemma 2.3.11. Let H be a multiplicatively closed subset of $M_{m\times m}(\mathbb{Z}[t]/(\hat{f}))$, and suppose that for every $\alpha \in H$, $\mathrm{Trace}(\alpha^{p^k})$ is divisible by p^{k+1}, where $k = \lfloor \log_p m \rfloor$. Then a is nilpotent for all $a \in A$ such that $a = (\alpha \bmod p)$ for some $\alpha \in H$.

Lemma 2.3.12. Let $\alpha \in M_{m\times m}(\mathbb{Z}[t]/(\hat{f}))$ such that $(\alpha \bmod p)$ is nilpotent in A. Then for every $i \geq 0$,

$$\mathrm{Trace}(\alpha^{p^i}) \equiv 0 \pmod{p^{i+1}}.$$

Theorem 2.3.13. I_j is an ideal of A for every j such that $-1 \leq j \leq k$, and $I_k = \mathrm{rad}(A)$.

Theorem 2.3.14.

(i) The functions g_i are $\mathbb{F}_p$-linear on I_{i-1} for all i, $0 \leq i \leq k$.

(ii) $I_i = \{\, a \in I_{i-1} \,:\, g_i(ab) = 0 \text{ for all } b \in A \,\}$.

Theorems 2.3.13 and 2.3.14 establish the correctness of this method for computation of the radical. We add a final result which will be used to make the resulting algorithm (shown on the next page) more efficient.

Lemma 2.3.15. If $a \in I_{i-1}$, and if $g_i(a) = 0$, then $g_i(\gamma a) = 0$ for all $\gamma \in \mathbb{F}_{p^l}$.

Proof. Let $\alpha \in M_{m\times m}(\mathbb{Z}[t]/(\hat{f}))$ and let $\hat{\gamma} \in \mathbb{Z}[t]/(\hat{f})$ so that $(\alpha \bmod p) = a$ and $(\hat{\gamma} \bmod p) = \gamma$. Then

$$\begin{aligned} g_i(\gamma a) &= (\hat{g}_i(\hat{\gamma}\alpha)) \pmod p \\ &= \left(\frac{1}{p^i}\,\mathrm{Trace}((\hat{\gamma}\alpha)^{p^i})\right) \pmod p \\ &= \left(\hat{\gamma}^{p^i} \bmod p\right)\left(\frac{1}{p^i}\,\mathrm{Trace}(\alpha^{p^i}) \bmod p\right) \\ &= (\gamma^{p^i})g_i(a) = 0, \qquad \text{if } g_i(a) = 0. \quad \blacksquare \end{aligned}$$

We assume $\mathbb{F}_{p^l}$ is represented as $\mathbb{F}_p[t]/(f)$ for an irreducible polynomial $f \in \mathbb{F}_p[t]$ in this algorithm. However, the method can be used to isolate the radical of a finite-dimensional algebra A over $\mathbb{F}_{p^l}$, with elements of $\mathbb{F}_{p^l}$ represented as $\mathbb{F}_p$-linear combinations of elements of any basis $\gamma_1, \gamma_2, \ldots, \gamma_l$ for $\mathbb{F}_{p^l}$ over $\mathbb{F}_p$ — provided that a multiplication table for this basis (which includes the elements $a_{i\,j\,k}$ of $\mathbb{F}_p$, for $1 \leq i, j, k \leq l$, with $\gamma_i \cdot \gamma_j = \sum_{k=1}^{l} a_{i\,j\,k}\gamma_k$) is given with the description of the algebra A. As stated, the algorithm uses the basis

$$1 + (f),\ t + (f),\ \ldots,\ t^{l-1} + (f)$$

for $\mathbb{F}_{p^l}$ over $\mathbb{F}_p$; the coefficients of f replace a multiplication table for this basis, in the input.

Algorithm **Isolation of the Radical — Positive Characteristic**

Input.

- Integers n, m, p, $l > 0$, with p prime.
- Coefficients f_{l-1}, f_{l-2}, $\ldots$, f_1, f_0 of a monic irreducible polynomial
  $f = t^l + f_{l-1}t^{l-1} + \cdots + f_1 t + f_0 \in \mathbb{F}_p[t]$.
- Matrices $a_1, a_2, \ldots, a_n \in M_{m\times m}(\mathbb{F}_p[t]/(f))$, which form the basis of a finite-dimensional associative algebra $A \subseteq M_{m\times m}(\mathbb{F}_p[t]/(f))$, with each entry (α) of each matrix represented by the coefficients of a polynomial $(\hat{\alpha})$ in $\mathbb{F}_p[t]$ with degree less than l (such that $\alpha = (\hat{\alpha} \bmod f)$).

Output.

- Integer $r \geq 0$, the dimension of $\mathrm{rad}(A)$ over $\mathbb{F}_p[t]/(f)$.
- Elements $\mu_{i\,j}$ of $\mathbb{F}_p[t]/(f)$, with $b_i = \sum_{j=1}^{n} \mu_{i\,j}\, a_j$, so that
  (1) $b_1, b_2, \ldots, b_r$ is a basis for $\mathrm{rad}(A)$ over $\mathbb{F}_p[t]/(f)$;
  (2) $b_1, b_2, \ldots, b_n$ is a basis for A over $\mathbb{F}_p[t]/(f)$.
- Matrices $c_1, c_2, \ldots, c_r \in M_{(n-r)\times(n-r)}(\mathbb{F}_p[t]/(f))$ forming the basis for a semi-simple associative algebra over $\mathbb{F}_p[t]/(f)$ which is isomorphic to the factor algebra $A/\mathrm{rad}(A)$.

```
(1)  Form a basis $\{ c_1, c_2, \ldots, c_{nl} \}$ for $I_{-1} = A$ over $\mathbb{F}_p$.
     for $i = 0, 1, \ldots, \lfloor \log_p m \rfloor$
         (Suppose $\{ d_1, d_2, \ldots, d_s \}$ is a basis for $I_{i-1}$ over $\mathbb{F}_p$.)
(2)      Compute the coefficient matrix (over $\mathbb{F}_p$) for the system of
         equations (in indeterminates $\lambda_1, \lambda_2, \ldots, \lambda_s$)
             $g_i((\lambda_1 d_1 + \lambda_2 d_2 + \cdots + \lambda_s d_s)\, a_j) = 0$,
         for $1 \leq j \leq n$ and for $g_i$ as defined on page 67.
(3)      Compute a basis (over $\mathbb{F}_p$) for the set of solutions of this system.
         Use this to generate a basis for the set $I_i$ over $\mathbb{F}_p$, such that
         $\lambda_1 a_1 + \lambda_2 a_2 + \cdots \lambda_n a_n \in I_i$ if and only if $(\lambda_1, \lambda_2, \ldots, \lambda_n)$
         is a solution of the system of equations defined in step 2.
     end for
(4)  Use the basis for $I_k = \mathrm{rad}(A)$ over $\mathbb{F}_p$ (for $k = \lfloor \log_p m \rfloor$) to generate
     a basis $b_1, b_2, \ldots, b_r$ for $\mathrm{rad}(A)$ over $\mathbb{F}_p[t]/(f)$.
(5)  Extend this basis to obtain a basis $\{ b_1, b_2, \ldots, b_n \}$ for $A$ over $\mathbb{F}_p[t]/(f)$
     by adding each element $a_i$ which is not a linear combination of the
     elements $b_1, b_2, \ldots, b_r, a_1, a_2, \ldots, a_{i-1}$, for $1 \leq i \leq n$.
(6)  For $1 \leq i, j \leq n$, define $\mu_{i\,j} \in \mathbb{F}_p[t]/(f)$ such that
         $b_i = \mu_{i\,1} a_1 + \mu_{i\,2} a_2 + \cdots + \mu_{i\,n} a_n$, for $1 \leq i \leq n$.
(7)  Compute a set of structure constants for the algebra $A/\mathrm{rad}(A)$ with
     respect to the basis $b_{r+1} + \mathrm{rad}(A), b_{r+2} + \mathrm{rad}(A), \ldots, b_n + \mathrm{rad}(A)$,
     and set $c_i = \phi(b_{r+i} + \mathrm{rad}(A))$, for $1 \leq i \leq n - r$, and for $\phi$ the
     regular representation for $A/\mathrm{rad}(A)$ with respect to this basis.
```

Example 2.3.16. Suppose A is the algebra of 2×2 upper triangular matrices over $\mathbb{F}_4$, with basis $\{ a_1, a_2, a_3 \}$ over $\mathbb{F}_4$, for

$$a_1 = \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix}, \qquad a_2 = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \qquad a_3 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}.$$

We will use our method to compute the radical of A. We first note that $A \subseteq M_{2\times 2}(\mathbb{F}_4)$ and $p = \text{char } \mathbb{F}_4 = 2$; hence $k = \lfloor \log_p m \rfloor = \lfloor \log_2 2 \rfloor = 1$.

We use the isomorphism $\mathbb{F}_4 \cong \mathbb{F}_2[t]/(f)$, for $f = t^2 + t + 1$, when performing computations over $\mathbb{F}_2$. Let $\alpha \in \mathbb{F}_4$ such that $\alpha^2 + \alpha + 1 = 0$. We have a basis $\{ 1, \alpha \}$ for $\mathbb{F}_4$ over $\mathbb{F}_2$, and a basis

$$\{ a_1, \alpha a_1, a_2, \alpha a_2, a_3, \alpha a_3 \}$$

for $I_{-1} = A$ over $\mathbb{F}_2$.

We now compute a basis for I_0 over $\mathbb{F}_2$. This ideal is defined as the set of elements

$$z = t_1 a_1 + t_2 \alpha a_1 + t_3 a_2 + t_4 \alpha a_2 + t_5 a_3 + t_5 \alpha a_3$$

such that $t_1, t_2, t_3, t_4, t_5, t_6 \in \mathbb{F}_2$ and

$$g_0(z \cdot a_i) = 0 \qquad \text{and} \qquad g_0(z \cdot \alpha a_i) = 0 \qquad \text{for } 1 \le i \le 3.$$

By Lemma 2.3.15, and as indicated in step 2 of the algorithm, it is sufficient to check conditions

$$g_0(z \cdot a_i) = 0, \qquad\qquad \text{for } 1 \le i \le 3,$$

since these conditions imply the others.

Since the map g_0 is simply the trace, these are equivalent to the conditions

$$\text{Trace}\left(\begin{bmatrix} t_1 + t_2\alpha & (t_1 + t_3) + (t_2 + t_4)\alpha \\ 0 & t_5 + t_6\alpha \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix} \right) = 0 + 0\alpha,$$
$$\text{Trace}\left(\begin{bmatrix} t_1 + t_2\alpha & (t_1 + t_3) + (t_2 + t_4)\alpha \\ 0 & t_5 + t_6\alpha \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \right) = 0 + 0\alpha,$$
$$\text{Trace}\left(\begin{bmatrix} t_1 + t_2\alpha & (t_1 + t_3) + (t_2 + t_4)\alpha \\ 0 & t_5 + t_6\alpha \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \right) = 0 + 0\alpha,$$

for $t_1, t_2, t_3, t_4, t_5, t_6 \in \mathbb{F}_2$. These are equivalent to the conditions

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} t_1 \\ t_2 \\ t_3 \\ t_4 \\ t_5 \\ t_6 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}.$$

Solving this system, we see that it is equivalent to the condition $t_1 = t_2 = t_5 = t_6 = 0$. Thus $\{ a_2, \alpha a_2 \}$ is a basis for I_0 over $\mathbb{F}_2$.

We next compute a basis for I_1 over $\mathbb{F}_2$. This is the set of elements $t_1a_2+t_2\alpha a_2 \in I_0$ such that $t_1,\, t_2 \in \mathbb{F}_2$, and

$$\begin{aligned}
&\text{(i)} \quad g_1((t_1a_2 + t_2\alpha a_2) \cdot a_1) = 0;\\
&\text{(ii)} \quad g_1((t_1a_2 + t_2\alpha a_2) \cdot a_2) = 0;\\
&\text{(iii)} \quad g_1((t_1a_2 + t_2\alpha a_2) \cdot a_3) = 0.
\end{aligned}$$

Now

$$((t_1a_2 + t_2\alpha a_2) \cdot a_1) = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix},$$

$$((t_1a_2 + t_2\alpha a_2) \cdot a_2) = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix},$$

and

$$((t_1a_2 + t_2\alpha a_2) \cdot a_3) = \begin{bmatrix} 0 & t_1 + t_2\alpha \\ 0 & 0 \end{bmatrix}.$$

Let $\hat{\alpha} \in \mathbb{Z}[t]/(\hat{f})$ so that $(\hat{\alpha} \bmod 2) = \alpha$, so $\hat{\alpha}^2 = -\hat{\alpha} - 1$. Now

$$\begin{aligned}
g_1((t_1a_2 + t_2\alpha a_2) \cdot a_3) &= \hat{g}_1\left(\begin{bmatrix} 0 & t_1 + t_2\hat{\alpha} \\ 0 & 0 \end{bmatrix}\right) \bmod 2\\
&= \left[\frac{1}{2}\,\text{Trace}\left(\begin{bmatrix} 0 & t_1 + t_2\hat{\alpha} \\ 0 & 0 \end{bmatrix}^2\right)\right] \bmod 2\\
&= 0 \bmod 2 = 0.
\end{aligned}$$

Similarly, $g_1((t_1a_2+t_2\alpha a_2)\cdot a_1) = 0$ and $g_1((t_1a_2+t_2\alpha a_2)\cdot a_2) = 0$. Thus I_1 contains the values $t_1a_2 + t_2\alpha a_2$ such that $t_1,\, t_2 \in \mathbb{F}_2$, and

$$\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} t_1 \\ t_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}.$$

We see that $\{ a_2,\, \alpha a_2 \}$ is a basis for $I_1 = \text{rad}(A)$ over $\mathbb{F}_2$; thus $\{ a_2 \}$ is a basis for $\text{rad}(A)$ over $\mathbb{F}_4$.

We extend this to a basis $\{ b_1,\, b_2,\, b_3 \}$ for A over $\mathbb{F}_4$ with $b_1 = a_2$, $b_2 = a_1$, and $b_3 = a_3$, and compute a regular matrix representation for $A/\text{rad}(A)$, as in Example 2.3.5.

We could continue by computing a basis for the radical of the algebra A of Example 2.3.16 from structure constants for A (as in Example 2.3.6). We would find that the function g_1 obtained in this computation is $\mathbb{F}_2$-linear, but not $\mathbb{F}_4$-linear, on the set I_0 — and that the set $I_0 \neq \mathrm{rad}(A)$ for this example.

Theorem 2.3.17. Let $A \subseteq M_{m\times m}(F)$ be a finite-dimensional associative algebra of dimension n over a field $F = \mathbb{F}_{p^l}$ for $m > 0$, prime $p > 0$, and for $l > 0$. Suppose we are given a basis for A over F, and the coefficients of a monic irreducible polynomial $f \in \mathbb{F}_p[t]$, with degree l.

(i) The output of the problem "Isolation of the Radical" can be computed using arithmetic over F, using a number of arithmetic steps which is polynomial in nml.

(ii) If elements of $F = \mathbb{F}_p[t]/(f) \cong \mathbb{F}_{p^l}$ are represented as vectors of elements of $\mathbb{F}_p$, with each $\alpha \in \mathbb{F}_p[t]/(f)$ represented by the coefficients of a polynomial $\hat{\alpha} \in \mathbb{F}_p[t]$ with degree less than l such that $\alpha = (\hat{\alpha} \bmod f)$, then the output of the problem "Isolation of the Radical" can be computed using arithmetic over $\mathbb{F}_p$ using a number of steps polynomial in nml, or in parallel using arithmetic-Boolean circuits over the field $\mathbb{F}_p$ of size polynomial in nml and of depth $O(\log^2(nml)\log_p m)$.

(iii) These outputs can be computed using a number of Boolean operations which is polynomial in $nml \log p$, or using Boolean circuits of size polynomial in $nml \log p$ and of depth $O(\log^2(nml \log p)\log_p m)$.

Proof. We first note that if $p > m$ then, since $\lfloor \log_p m \rfloor = 0$, Dickson's criterion can be used to compute a basis for the radical of A — that is, we can apply the algorithm (and the timing analysis for "arithmetic" steps) for fields of characteristic zero.

Suppose now that $p \leq m$, so $\lfloor \log_p m \rfloor \geq 1$. We first consider the arithmetic cost of computing the output for "Isolation of the Radical" sequentially. As noted by Lempel, Seroussi, and Ziv [80], we can assume that elements of $\mathbb{F}_{p^l}$ are represented as vectors of elements of $\mathbb{F}_p$, and that we perform arithmetic over this smaller field. We compute a basis for the radical by forming and solving $k = 1 + \lfloor \log_p m \rfloor$ systems of linear equations of polynomial size — and it is clear that this can be done in polynomial time. It is also clear that the number of Boolean operations used to compute this output is polynomial in the size of the input.

We next consider the cost of computing the desired output in parallel. We first consider the case $l = 1$ — so $F = \mathbb{F}_{p^l} = \mathbb{F}_p$. The bounds stated in (ii) and (iii) follow from well known bounds for the parallel cost of solving systems of linear equations (and from the fact that p is small: $p \leq m$). These bounds can also be attained for arbitrary l by using an efficient implementation of arithmetic over finite extensions of $\mathbb{F}_p$ (See Section 1.3 for details). ■

Thus the restriction of Friedl and Rónyai's algorithm to computation over prime fields is not strictly necessary — it can be applied "directly" to compute the radical of algebras over $\mathbb{F}_{p^l}$. We can prove a slightly better upper bound on running time than Friedl and Rónyai (note that we use $k = 1 + \lfloor \log_p m \rfloor$, rather than $1 + \lfloor \log_p ml \rfloor$). Unfortunately, we have been unable to generalise the algorithm beyond that: it requires the solution of systems of $\mathbb{F}_p$-linear equations which are not linear over the ground field. Hence it requires arithmetic over $\mathbb{F}_p$, and the assumption that the ground field be a finite extension of $\mathbb{F}_p$.

2.3.3. Algebras without Identity Elements

We noted in Section 2.1 that Friedl and Rónyai [43] do not require "associative algebras" to have multiplicative identities. Suppose now that $\hat{A}$ is a vector space of degree n over a field F which satisfies the definition of "associative algebra over F" given by Friedl and Rónyai, and which does not have a multiplicative identity.

As noted in Example 2.1.7, the set

$$A = \{\, \alpha 1_A + a \;:\; \alpha \in F, a \in \hat{A} \,\}$$

is an associative algebra of dimension $n+1$ over F (with multiplication in $\hat{A}$ extended to A as shown in that example). Suppose now that $x = \alpha 1_A + a$ is nilpotent in A, with $\alpha \in F$ and $a \in \hat{A}$; then, since $\hat{A}$ is a two-sided ideal of A, and $x^k = \alpha^k 1_A + a_k$ for some $a_k \in \hat{A}$, it is clear that $\alpha = 0$ and $x \in \hat{A}$. Thus every strongly nilpotent element of A is in $\hat{A}$, and every nilpotent (left) ideal in A is a nilpotent (left) ideal contained in $\hat{A}$. Clearly (for our definition of the radical, or for the definition used by Friedl and Rónyai),

$$\text{rad}(A) = \text{rad}(\hat{A}).$$

Thus the algorithms given by Friedl and Rónyai can be used to isolate the radical of a finite-dimensional associative algebra, using either definition of "associative algebra".

Now, since $\text{rad}(A)$ is a two-sided ideal contained in $\hat{A}$,

$$A/\text{rad}(A) \cong \{\, \alpha 1_A + \hat{a} \;:\; \alpha \in F, \hat{a} \in \hat{A}/\text{rad}(\hat{A}) \,\},$$

so that $A/\text{rad}(\hat{A})$ is a semi-simple associative algebra over F with dimension one greater than the dimension of the F-vector space $\hat{A}/\text{rad}(\hat{A})$; again, this vector space is an "associative algebra", as defined by Friedl and Rónyai.

The structure theorems (Theorems 2.1.21 and 2.1.22) are correct for "associative algebras" without identity. In fact, they imply that any finite-dimensional semi-simple "associative algebra" over F, including $\hat{A}/\text{rad}(\hat{A})$, is isomorphic to a direct sum of simple algebras over F *with* identity elements. Either $\hat{A} = \text{rad}(\hat{A})$, and

$\hat{A}/\text{rad}(\hat{A}) = (0)$, or $\hat{A}/\text{rad}(\hat{A})$ is a semi-simple associative algebra over F (with identity). For a more detailed discussion of these properties of associative algebras (with or without identity elements), see Chapter 13 of van der Waerden [118].

It follows that we can ignore the distinction between our definition of "associative algebra" and the definition used by Friedl and Rónyai in Sections 2.4 and 2.5, since these definitions are equivalent for finite-dimensional semi-simple algebras.

The following example shows that the extreme case "$A = \text{rad}(A)$" can occur if the definitions of Friedl and Rónyai are used.

Example 2.3.18. Consider the ring of strictly upper triangular matrices over a field F,

$$\hat{A} = \{\, U = (U_{i\,j})_{1 \le i,\, j \le n} \in M_{n\times n}(F) \;:\; U_{i\,j} = 0 \text{ if } j \le i \,\}.$$

It is easily checked that for $k > 0$,

$$\hat{A}^k = \{\, U = (U_{i\,j})_{1 \le i,\, j \le n} \in M_{n\times n}(F) \;:\; U_{i\,j} = 0 \text{ if } j \le i + k - 1 \,\}.$$

In particular, $\hat{A}^{n+1} = 0$. Thus $\hat{A}$ is a nilpotent ideal of itself; so $\text{rad}(\hat{A}) = \hat{A}$, and $\hat{A}/\text{rad}(\hat{A}) = (0)$.

2.3.4. Algebras over Field Extensions

As stated in Section 2.2, we are also interested in the decomposition of an algebra $A_E = A \otimes_F E$ over a field E, given an algebra A of dimension n over a field F, and an extension E of F. We will be interested in the case that F is a finite algebraic extension of $\mathbb{Q}$ or a finite field, and that E is an algebraic extension of F. We begin with a result about the structure of the radical of $A_E = A \otimes_F E$, for E a separable extension of a field F.

Proposition 2.3.19. Let A be a finite-dimensional associative algebra of dimension n over a field F, and suppose $\text{rad}(A)$ has dimension r and basis $b_1, b_2, \ldots, b_r$ over F. Let E be a finite separable extension of F. Then the radical of A_E has dimension r and basis $b_1, b_2, \ldots, b_r$ over E.

For a proof of this, see Section 69 of Curtis and Reiner [31]. We use it to prove the following theorem.

Theorem 2.3.20. Let A be a finite-dimensional associative algebra of dimension n over a field F. Suppose the radical of A has dimension r and basis $b_1, b_2, \ldots, b_r$ over F.

(i) If F has characteristic zero and E is any extension of F then the algebra A_E has a radical of dimension r and with basis $b_1, b_2, \ldots, b_r$ over E.

(ii) If F is a perfect field and E is any algebraic extension of F then the algebra A_E has a radical of dimension r and with basis $b_1, b_2, \ldots, b_r$ over E.

(iii) There exist fields F and E, with E an algebraic extension of dimension 2 over F, and a finite-dimensional associative algebra A of dimension 2 over F, such that $\mathrm{rad}(A) = (0)$, but $\mathrm{rad}(A_E) \neq (0)$.

Proof. Part (i) follows from the fact that Dickson's criterion for membership in the radical of A is correct for finite-dimensional algebras over fields of characteristic zero. Given a basis $a_1, a_2, \ldots, a_n$ for A over F, we use Dickson's criterion to construct a matrix Z with entries in F such that, for $\lambda_1, \lambda_2, \ldots, \lambda_n \in F$,

$$\lambda_1 a_1 + \lambda_2 a_2 + \cdots + \lambda_n a_n \in \mathrm{rad}(A) \qquad \text{if and only if} \qquad Z \cdot \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_n \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

Applying Dickson's criterion to construct a basis for $\mathrm{rad}(A_E)$ over E, using the basis $a_1, a_2, \ldots, a_n$ for A_E over E, we obtain the same coefficient matrix Z. Since the rank of Z is independent of the choice of ground field (between F and E), the result follows.

We use Proposition 2.3.19 to prove part (ii) of the theorem. Suppose F, E, and A are as stated above, and $b_1, b_2, \ldots, b_r$ is a basis for $\mathrm{rad}(A)$ over F. Now $\mathrm{rad}(A)$ is a nilpotent ideal spanned (over F) by $b_1, b_2, \ldots, b_r$: $(\mathrm{rad}(A))^k = (0)$ for some $k \geq 0$. Let $I \subseteq A_E$ be the E-vector space spanned by $b_1, b_2, \ldots, b_r$. It is easily checked that I is a two-sided ideal of A_E and that $I^k = (0)$. Thus I is a nilpotent ideal of A_E, and it follows by Theorem 2.1.12 that $I \subseteq \mathrm{rad}(A_E)$.

Suppose $\mathrm{rad}(A_E) \nsubseteq I$; then there exists some $\alpha \in A_E$ such that $\alpha \in \mathrm{rad}(A_E) \setminus I$. However, there exists some finite algebraic extension $\hat{E}$ of F such that $F \subseteq \hat{E} \subseteq E$ and $\alpha \in A \otimes_F \hat{E}$ (This follows easily from the fact that A_E is isomorphic to a subring of $M_{n\times n}(E)$ and that E is an algebraic extension of F). Now it is clear that $\alpha \in \mathrm{rad}(A \otimes_F \hat{E})$ — hence (by Proposition 2.3.19), α is in the $\hat{E}$-subspace spanned by $b_1, b_2, \ldots, b_r$, contradicting the fact that $\alpha \notin I$. Thus $\mathrm{rad}(A_E) = I$, proving part (ii).

To prove part (iii), we consider the field $F = \mathbb{F}_2(t)$ of rational functions in one indeterminate over $\mathbb{F}_2$. Let A be the algebra over F spanned by the matrices

$$a_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \qquad a_2 = \begin{bmatrix} 0 & t \\ 1 & 0 \end{bmatrix}.$$

Now $a_2^2 = ta_1$, so we see that A is a finite-dimensional associative algebra of dimension 2 over F. To see that $\mathrm{rad}(A) = (0)$, let $\gamma_1, \gamma_2 \in F$ such that $\gamma_1 a_1 + \gamma_2 a_2 \in \mathrm{rad}(A)$. Then $\gamma_1 a_1 + \gamma_2 a_2$ is nilpotent, so it is clear that the matrix

$$\gamma_1 a_1 + \gamma_2 a_2 = \begin{bmatrix} \gamma_1 & t\gamma_2 \\ \gamma_2 & \gamma_1 \end{bmatrix}$$

is singular: $\det(\gamma_1 a_1 + \gamma_2 a_2) = \gamma_1^2 + t\gamma_2^2 = 0$. Since $\gamma_1, \gamma_2 \in \mathbb{F}_2(t)$, it is clear that $\gamma_1 = \gamma_2 = 0$.

Now let $E = F[x]/(x^2 + t)$. E is a finite algebraic extension of dimension 2 over F. Consider the element $xa_1 + a_2$ of A_E; $xa_1 + a_2 \neq 0$, but

$$(xa_1 + a_2)^2 = \begin{bmatrix} x & t \\ 1 & x \end{bmatrix}^2 = \begin{bmatrix} x^2 + t & tx + tx \\ x + x & t + x^2 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix},$$

so $(xa_1 + a_2)$ is nilpotent. Since A_E is a commutative algebra, it is clear that this element is also strongly nilpotent. Hence $(xa_1 + a_2) \in \mathrm{rad}(A_E) \neq (0)$, proving part (iii). ■

We now have an algorithm for the (arithmetic or Boolean) computation of the radical of a finite-dimensional associative algebra A over any field F which is an algebraic extension of $\mathbb{Q}$ or of a prime field $\mathbb{F}_p$. Given a basis $a_1, a_2, \ldots, a_n$ and set of structure constants $\gamma_{i\,j\,k}$ (for $1 \leq i, j, k \leq n$) for A over F, we note that there exists a field $\hat{F}$ containing these structure constants, such that $\hat{F} \subseteq F$ and $\hat{F}$ is a *finite* algebraic extension of $\mathbb{Q}$, or of $\mathbb{F}_p$ (and in the latter case, $\hat{F}$ is a finite field). We compute the minimal polynomial of a generator α of $\hat{F}$ over $\mathbb{Q}$ or $\mathbb{F}_p$ (as well as an isolating rectangle for α, if $\hat{F}$ is an extension of $\mathbb{Q}$), and compute a basis over $\hat{F}$ for the radical of the algebra $\hat{A}$ with basis $a_1, a_2, \ldots, a_n$ over $\hat{F}$ and structure constants $\gamma_{i\,j\,k}$. By Theorem 2.3.20 (ii), this also gives us a basis for $\mathrm{rad}(A)$ over F.

We also have a means of computing a basis for the radical of $A \otimes_F \mathbb{R}$ or $A \otimes_F \mathbb{C}$ for any number field F and finite-dimensional associative algebra A over F: We simply use Dickson's criterion to compute (and return as output) a basis for the radical of A over F.

Having considered algebraic extensions of $\mathbb{Q}$ and of finite fields, it seems natural to consider the next simplest set of fields — those of the form $G(x_1, x_2, \ldots, x_m)$, for G a finite algebraic extension of $\mathbb{Q}$ or of a finite field, and for indeterminates $x_1, x_2, \ldots, x_n$ over G. If G has characteristic zero, then the algorithm based on Dickson's criterion can be used to compute a basis for the radical of a finite-dimensional associative algebra A over $F = G(x_1, x_2, \ldots, x_n)$, using a polynomial number of field operations over the field F. However, it is not clear that the algorithm for "Isolation of the Radical" of algebras over finite fields can be generalised.

Question 2.3.21. What is the complexity of the problem "Isolation of the Radical", for finite-dimensional associative algebras over fields $\mathbb{F}_{p^l}(x_1, x_2, \ldots, x_m)$?

2.3.5. Computation of the Radical and Squarefree Decomposition

We conclude this chapter by considering computations of the radical for finite-dimensional associative algebras over arbitrary fields. We relate this to a better known problem — the squarefree decomposition of polynomials, discussed in Section 1.4.

Theorem 2.3.22. Suppose a basis for the radical of an arbitrary associative algebra of dimension n over an field F can be computed from a basis and set of structure constants, using arithmetic-Boolean circuits over F of depth $D(n)$ and size $S(n)$. Then the monotone squarefree decomposition and distinct power decomposition of an arbitrary polynomial $f \in F[x]$ of degree n over F can be computed from the coefficients of f, using arithmetic-Boolean circuits of depth $D(n) + O(\log^2 n)$ and size $S(n) + n^{O(1)}$.

Proof. Consider the algebra $A = F[x]/(f)$ of dimension n over F. We have a basis $1 + (f)$, $x + (f)$, $x^2 + (f)$, $\ldots$, $x^{n-1} + (f)$ for A over F; the image $\phi(x + (f))$ of $x + (f)$ under the regular representation of A for this basis is simply the companion matrix of f (see Example 2.2.6). The matrices $\phi(x^i + (f))$, $0 \leq i < n$, can be computed from the coefficients of f using circuits of size $n^{O(1)}$ and depth $O(\log^2 n)$. Hence the structure constants for A with respect to this basis can be computed at this cost. Applying the size and depth bounds given for "Isolation of the Radical", we conclude that a basis for the radical of A over F can be computed from the coefficients of f using circuits of depth $D(n) + O(\log^2 n)$ and size $S(n) + n^{O(1)}$.

As claimed in Example 2.1.12, $\text{rad}(A) = (g_1)/(f)$, for g_1 the greatest squarefree part of f. To prove this, we note that A is a commutative algebra — hence an element $\alpha = a + (f)$ (for $a \in F[x]$) is in $\text{rad}(A)$ if and only if α is nilpotent in A. That is, $\alpha = a + (f) \in \text{rad}(A)$ if and only if f divides a^k in $F[x]$ for some $k \geq 0$. It is clear that this is the case if and only if g_1 divides a — that is, if $a \in (g_1)$, and $\alpha \in (g_1)/(f)$. Thus $\text{rad}(A) = (g_1)/(f)$.

Now suppose $b_1, b_2, \ldots, b_r$ is a basis for $\text{rad}(A)$ over F. We obtain the coefficients of g_1 from this basis by finding the smallest $k \geq 0$ such that $x^k + (f)$ is an F-linear combination of $b_1, b_2, \ldots, b_r$, $1 + (f)$, $x + (f)$, $\ldots$, $x^{k-1} + (f)$ — that is, such that there exist $\gamma_0, \gamma_1, \ldots, \gamma_{k-1} \in F$ with

$$x^k \equiv \gamma_{k-1}x^{k-1} + \gamma_{k-2}x^{k-2} + \cdots + \gamma_1 x + \gamma_0 \pmod{\text{rad}(A)}$$

in A — and by computing these coefficients $\gamma_{k-1}, \gamma_{k-2}, \ldots, \gamma_1, \gamma_0$. It is clear that

$$g_1 = x^k - \gamma_{k-1}x^{k-1} - \gamma_{k-2}x^{k-2} - \cdots - \gamma_1 x - \gamma_0.$$

The degree k and the coefficients of g_1 can be computed as described above, by solving n systems of linear equations (corresponding to the n possible values of k, $1 \leq k \leq n$) in parallel.

Finally, we note that the remaining polynomials in the monotone squarefree decomposition and distinct power decomposition of f can be computed from the coefficients of f and g_1 using the relationships stated in Proposition 1.4.6, using arithmetic-Boolean circuits of depth $O(\log^2 n)$ and size $n^{O(1)}$. Thus these decompositions of f can be computed at the stated cost. ■

As we noted in Section 1.4, there exist fields F for which the problem of deciding whether a polynomial $f \in F[x]$ is squarefree in $F[x]$ is undecidable. This is sufficient for us to conclude that no algorithm exists for deciding whether a finite-dimensional associative algebra is semi-simple, or for the computation of a basis for the radical of a finite-dimensional associative algebra, over an arbitrary field.

The reduction from the computation of the monotone squarefree decomposition and distinct power decomposition of polynomials to "Isolation of the Radical" (Theorem 2.3.22) cannot be used to resolve Question 2.3.21 — at least, it cannot be used to show that "Isolation of the Radical" is a hard problem for fields $\mathbb{F}_{p^l}(x_1, x_2, \ldots, x_m)$. There exist efficient algorithms for factorisation of polynomials in the polynomial ring $\mathbb{F}_{p^l}[x_1, x_2, \ldots, x_m]$, for extraction of the numerator and denominator $f, g \in \mathbb{F}_{p^l}[x_1, x_2, \ldots, x_m]$ given a representation of a rational function $\alpha = f/g \in \mathbb{F}_{p^l}(x_1, x_2, \ldots, x_m)$, and for determination of the degree in x_1 of a polynomial $f \in \mathbb{F}_{p^l}[x_1, x_2, \ldots, x_m]$, using a very general representation of multivariate polynomials in $\mathbb{F}_{p^l}[x_1, x_2, \ldots, x_m]$ (see Kaltofen [70], [71] for details). It is clear that these can be used to compute the monotone squarefree decomposition of a polynomial of degree n with coefficients in $\mathbb{F}_{p^l}(x_1, x_2, \ldots, x_m)$ efficiently. Thus a reduction from "Isolation of the Radical" over a field F to the computation of the monotone squarefree decomposition of polynomials with coefficients in F of the type stated in Theorem 2.3.22 would imply that the problem "Isolation of the Radical" has an efficient solution for fields $\mathbb{F}_{p^l}(x_1, x_2, \ldots, x_m)$.

Question 2.3.23. Is the problem "Isolation of the Radical" (polynomial time) reducible to the computation of the monotone squarefree decomposition of polynomials, over arbitrary fields?

In summary, we have efficient algorithms for computation of a basis for the radical of a finite-dimensional associative algebra over a finite extension of $\mathbb{Q}$, and over a finite field. While we can also compute the radical of a finite-dimensional associative algebra over an arbitrary algebraic extension of $\mathbb{Q}$ or over an arbitrary algebraic extension of a finite field, we know that the decision problem “Is A semi-simple?” is undecidable for an arbitrary finite-dimensional associative algebra A over an arbitrary field. The computation of the monotone squarefree decomposition and distinct power decomposition of polynomials in $F[x]$ can be reduced to the solution of “Isolation of the Radical”; a reduction from “Isolation of the Radical” to the computation of monotone squarefree decompositions would yield efficient solutions for “Isolation of the Radical” over fields where no such solutions are known.

2.4. Computation of Simple Components

We consider algorithms for the computation of the simple components of a finite-dimensional semi-simple associative algebra A over a field F. As stated in Theorem 2.1.23,

$$A = B_1 \oplus B_2 \oplus \cdots \oplus B_k$$

for simple algebras $B_1, B_2, \ldots, B_k$ (the *simple components* of A over F), for some $k > 0$. Given a basis and set of structure constants for A over F, or a set of matrices forming a basis for the (matrix) algebra A, we compute bases and structure constants for each of the simple components.

Problem **Extraction of Simple Components**

Input
- Integers n, $m > 0$.
- Matrices $a_1, a_2, \ldots, a_n \in M_{m\times m}(F)$, which form the basis for a finite-dimensional semi-simple associative algebra $A \subseteq M_{m\times m}(F)$ of dimension n over F.

Output
- Integer $k > 0$, the number of simple components of A.
- Integers $n_1, n_2, \ldots, n_k > 0$, with $n_1 + n_2 + \cdots + n_k = n$, such that n_i is the dimension of simple component B_i of A.
- Elements $\mu_{i\,j\,l}$ of F, for $1 \le i \le k$, $1 \le j \le n_i$, and $1 \le l \le n$, defining elements $b_{i\,j} = \sum_{l=1}^{n} \mu_{i\,j\,l}\, a_l$ of A such that
  (1) $b_{i\,1}, b_{i\,2}, \ldots, b_{i\,n_i}$ is a basis for the simple component B_i over F, and hence
  (2) $b_{1\,1}, \ldots, b_{k\,n_k}$ is a basis for A over F.
- Matrices $c_{i\,j} \in M_{n_i\times n_i}(F)$, such that $c_{i\,1}, c_{i\,2}, \ldots, c_{i\,n_i}$ is the basis for a matrix algebra isomorphic to B_i, for $1 \le i \le k$.

Friedl and Rónyai [43] show that this problem can be solved efficiently (using Boolean computations) when F is a finite field or a number field. Following their example, we begin (in Section 2.4.1) by reducing the above problem to the simpler problem of computing a set of central primitive idempotents for a semi-simple associative algebra. We show that this reduction is correct for a (slightly) more general class of fields than that discussed by Friedl and Rónyai, and for parallel Boolean and arithmetic computations. We continue by reviewing their algorithm (in Section 2.4.2), and by introducing two new algorithms for this problem (in Sections 2.4.3 and 2.4.4). The new algorithms eliminate the use of computations over extensions of the ground field, and reduce the use of factorisation of polynomials. Hence, we believe that they may be used to compute simple components of a semi-simple algebra more quickly than Friedl and Rónyai's algorithm. They also provide reductions from the computation of the simple components of semi-simple algebras over F to the factorisation of polynomials over F, for parallel Boolean computations (when F is a finite field or a number field) and for parallel arithmetic computations (when F is a perfect field). Finally, we present polynomial-time algorithms for the computation of simple components of $A \otimes_F \mathbb{R}$ and $A \otimes_F \mathbb{C}$, for a finite-dimensional semi-simple algebra A over a number field F; these algorithms are presented in Section 2.4.5.

2.4.1. Computation of Simple Components using Idempotents

Friedl and Rónyai show that the computation of bases for the simple components of a finite-dimensional, semi-simple associative algebra A over a field F can be reduced (with respect to polynomial-time computations) to the simpler problem of computation of the central primitive idempotents of A. In this section we review this reduction and show that it is also useful for parallel computations.

Definition 2.4.1. The *centre* of an associative algebra A over a field F, Centre(A), is the set of all elements of A which commute with all the elements in A:

$$\text{Centre}(A) = \{\, x \in A : xy = yx \text{ for all } y \in A \,\}.$$

Clearly Centre(A) is a commutative subalgebra of A. A basis for the centre of A over F can be computed by solving the system of linear equations

$$(\lambda_1 a_1 + \lambda_2 a_2 + \cdots + \lambda_n a_n) a_i = a_i (\lambda_1 a_1 + \lambda_2 a_2 + \cdots + \lambda_n a_n)$$

for $1 \le i \le n$, for indeterminates $\lambda_1, \lambda_2, \ldots, \lambda_n$ over F; the centre clearly includes every element $\lambda_1 a_1 + \lambda_2 a_2 + \ldots \lambda_n a_n$ of A such that $(\lambda_1, \lambda_2, \ldots, \lambda_n)$ is a solution of this system. The following facts indicate the usefulness of Centre(A) for the problem at hand.

Proposition 2.4.2. Let A be a finite-dimensional semi-simple associative algebra over a field F, such that $A = B_1 \oplus B_2 \oplus \cdots \oplus B_k$ for simple components $B_1, B_2, \ldots, B_k$ and for $k \ge 1$.
(i) $\text{Centre}(A) = \text{Centre}(B_1) \oplus \text{Centre}(B_2) \oplus \cdots \oplus \text{Centre}(B_k)$;
(ii) $B_i = A \cdot \text{Centre}(B_i) = \{\, \alpha\beta \;:\; \alpha \in A \text{ and } \beta \in \text{Centre}(B_i) \,\}$ for $1 \le i \le k$.

Proof. Let $\alpha \in A$; then $\alpha = \alpha_1 + \alpha_2 + \cdots + \alpha_k$ for a unique set $\alpha_1, \alpha_2, \ldots, \alpha_k$ with $\alpha_i \in B_i$ for $1 \le i \le k$. Suppose now that $\gamma = \gamma_1 + \gamma_2 + \cdots + \gamma_k \in \text{Centre}(A)$, with $\gamma_i \in B_i$ for $1 \le i \le k$. Since $\alpha_i \gamma_j = \gamma_j \alpha_i = 0$ if $1 \le i, j \le k$ and $i \ne j$, it is clear that

$$\begin{aligned} \alpha\gamma &= \alpha_1\gamma_1 + \alpha_2\gamma_2 + \cdots + \alpha_k\gamma_k, \\ \gamma\alpha &= \gamma_1\alpha_1 + \gamma_2\alpha_2 + \cdots + \gamma_k\alpha_k, \end{aligned}$$

and, since $\gamma \in \text{Centre}(A)$, $\alpha\gamma = \gamma\alpha$, and $\alpha_i\gamma_i$, $\gamma_i\alpha_i \in B_i$ for $1 \le i \le k$, it follows that $\alpha_i\gamma_i = \gamma_i\alpha_i$ for $1 \le i \le k$. Since α is arbitrarily chosen from A, it is clear that α_i is arbitrarily chosen from B_i, so $\gamma_i \in \text{Centre}(B_i)$. Thus $\text{Centre}(A) \subseteq \text{Centre}(B_1) \oplus \cdots \oplus \text{Centre}(B_k)$. It is also easily checked that if γ_i is arbitrarily chosen from $\text{Centre}(B_i)$ for $1 \le i \le k$, then $\alpha(\gamma_1 + \cdots + \gamma_k) = (\gamma_1 + \cdots + \gamma_k)\alpha$ for arbitrary $\alpha \in A$. Thus it is also true that $\text{Centre}(A) \supseteq \text{Centre}(B_1) \oplus \cdots \oplus \text{Centre}(B_k)$, proving (i).

Since the multiplicative identity of B_i is in $\text{Centre}(B_i)$, and $B_i \subseteq A_i$, it is clear that $B_i \subseteq \text{Centre}(B_i) \cdot A$. On the other hand, $\text{Centre}(B_i) \subseteq B_i$, and B_i is a two-sided ideal of A; thus $\text{Centre}(B_i) \cdot A \subseteq B_i$ as well, as required to prove (ii). ■

As Friedl and Rónyai note, Proposition 2.4.2 gives a reduction from "Extraction of Simple Components" in the general case to the problem for commutative, finite-dimensional semi-simple algebras. As noted above, a basis for the centre of a finite-dimensional associative algebra can be computed by solving a system of linear equations. By Proposition 2.4.2 (i) and (ii), the simple components of A are easily computed from those of $\text{Centre}(A)$ as well.

Suppose again that A has simple components $B_1, B_2, \ldots, B_k$, and that B_i has multiplicative identity e_i, for $1 \le i \le k$. Then the elements $e_1, e_2, \ldots, e_k$ comprise a set of *central primitive idempotents* for A, as defined below.

Definition 2.4.3. Let A be an associative algebra over a field F. A set of elements $i_1, i_2, \ldots, i_k$ of A is a *set of idempotents* for A if

(i) $i_1 + i_2 + \cdots + i_k = 1$ in A;

(ii) $i_r i_s = \delta_{r\,s} i_r = \begin{cases} i_r & \text{if } r = s, \\ 0 & \text{otherwise,} \end{cases}$ for $1 \le r, s \le k$.

It is a set of *primitive* idempotents if, in addition,

(iii) Any idempotent i of A (with $i^2 = i$) is the sum of some subset of the idempotents $i_1, i_2, \ldots, i_k$.

It is a set of *central* idempotents if (i) and (ii) hold and if $i_r \in \text{Centre}(A)$ for $1 \le r \le k$. Finally, it is a set of *central primitive* idempotents if they comprise a set of central idempotents and every *central* idempotent i of A is the sum of some subset of the idempotents $i_1, i_2, \ldots, i_k$.

Note that a set of central primitive idempotents is not generally a set of primitive idempotents.

Since $e_i A = B_i$ for $1 \le i \le k$, it is clear that we can isolate the simple components of A if we can compute the central primitive idempotents $e_1, e_2, \ldots, e_k$. We state this reduction formally on the following page.

Algorithm **Simple Components via Central Primitive Idempotents**

Input.

- Integers n, m, $k > 0$.
- Matrices $a_1, a_2, \ldots, a_n \in M_{m\times m}(F)$, which form the basis for a finite-dimensional semi-simple associative algebra $A \subseteq M_{m\times m}(F)$ of dimension n over F, and with k simple components over F.
- The central primitive idempotents $e_1, e_2, \ldots, e_k$ of A.

Output.

- Integers $n_1, n_2, \ldots, n_k > 0$, with $n_1 + n_2 + \cdots + n_k = n$, such that n_i is the dimension of the simple component B_i of A with identity element e_i, for $1 \le i \le k$.
- Elements $\mu_{i\,j\,l}$ of F, for $1 \le i \le k$, $1 \le j \le n_i$, and $1 \le l \le n$, defining elements $b_{i\,j} = \sum_{l=1}^{n} \mu_{i\,j\,l} a_l$ of A such that
  (1) $b_{i\,1}, b_{i\,2}, \ldots, b_{i\,n_i}$ is a basis for the simple component B_i over F, and hence
  (2) $b_{1\,1}, \ldots, b_{k\,n_k}$ is a basis for A over F.
- Matrices $c_{i\,j} \in M_{n_i\times n_i}(F)$ such that $c_{i\,1}, c_{i\,2}, \ldots, c_{i\,n_i}$ is a basis for a matrix algebra isomorphic to B_i, for $1 \le i \le k$.

(1) For $1 \le i \le k$, compute the integer n_i and a basis $b_{i\,1}, b_{i\,2}, \ldots, b_{i\,n_i}$ over F for $B = e_i A$, by selecting a maximal linearly independent subset of $e_i a_1, e_i a_2, \ldots, e_i a_n$.

(2) Compute elements $\mu_{i\,j\,l}$ of F, for $1 \le i \le k$, $1 \le j \le n_i$, and $1 \le l \le n$, such that $b_{i\,j} = \mu_{i\,j\,1} a_1 + \mu_{i\,j\,2} a_2 + \cdots + \mu_{i\,j\,n} a_n$ (by forming and solving n nonsingular systems of linear equations, each in n indeterminates, over F).

(3) For $1 \le i \le k$, compute a set of structure constants for the simple algebra B_i with respect to the basis $b_{i\,1}, b_{i\,2}, \ldots, b_{i\,n_i}$ over F (by forming and solving n_i^2 nonsingular systems of linear equations, each in n_i indeterminates over F). Use these to compute the matrix
$c_{i\,j} = \phi_i(b_{i\,j}) \in M_{n_i\times n_i}(F)$, for $1 \le i \le k$ and $1 \le j \le n_i$,
and for ϕ_i the regular matrix representation of B_i with respect to the above basis.

Theorem 2.4.4. Let $A \subseteq M_{m \times m}(F)$ be a finite-dimensional semi-simple associative algebra of dimension n over a field F. Given a basis for A over F, and the central primitive idempotents of A, bases for the simple components of A over F can be computed using arithmetic-Boolean circuits of depth $O(\log^2(mn))$ and size $(mn)^{O(1)}$.

Proof. The algorithm "Simple Components via Central Primitive Idempotents" can be used to perform this computation. The algorithm is clearly correct, since each central primitive idempotent e_i is the identity element of a simple component B_i, which is itself a two-sided ideal of A. The timing analysis follows from the analysis given for solutions of nonsingular systems of linear equations, in Section 1.3. ■

We are left with the problem of computing the central primitive idempotents $e_1, e_2, \ldots, e_k$ of a semi-simple algebra A. We first consider the problem of deciding whether A is simple (and $k = 1$). As stated in Theorem 2.1.25, a finite-dimensional simple associative algebra over a field F is isomorphic to a matrix ring $M_{h \times h}(D)$, for $h > 0$ and for a division algebra D over F. If A is commutative then it is clear that $h = 1$, so A is isomorphic to D, and that D is a commutative division algebra. Hence A is a field, and a finite algebraic extension of F. Conversely, it is clear that if A is a field, then A is simple and commutative: If A is not simple then Centre(A) includes nonzero zero-divisors (for example, the central primitive idempotents).

We will see that for some fields F, the problem of deciding whether a finite-dimensional semi-simple algebra A over F is a simple algebra is undecidable. However, the fact that a commutative finite-dimensional simple algebra is also a field gives us a method for deciding this problem for perfect fields. We make use of the following fact.

Proposition 2.4.5. If F is a perfect field, and E is an algebraic extension of F with finite dimension over F, then E is a primitive algebraic extension. That is, $E = F[\alpha]$ for some $\alpha \in E$.

Proposition 2.4.5 is a consequence of the more general result that every extension $F(\alpha_1, \alpha_2, \ldots, \alpha_n)$ with α_i separable and algebraic over F for $1 \leq i \leq n$ is a primitive algebraic extension (see van der Waerden [117] for a proof of this). If F is perfect, then every element α_i of E which is algebraic over F is also separable, so Proposition 2.4.5 follows.

Recall that the *minimal polynomial* $f \in F[t]$ of an element a of A is the monic nonzero polynomial of least degree such that $f(a) = 0$. (Note that if A is finite-dimensional over F then some such polynomial exists.) The minimal polynomial of a is a divisor in $F[t]$ of any polynomial $g \in F[t]$ such that $g(a) = 0$. If A is not a field, then an element a of A can have a minimal polynomial which is reducible in $F[t]$: In particular, if a is an idempotent in A other than 0 or 1, then a has minimal polynomial $t(t-1)$.

Proposition 2.4.6. Let $a \in A$, with minimal polynomial $f \in F[t]$. Then $F[a] \subseteq A$, and $F[a]$ is a field if and only if f is irreducible in $F[t]$.

Clearly, $F[a] \cong F[t]/(f)$. If f is irreducible then every nonzero element of $F[t]/(f)$ has a multiplicative inverse (which can be computed using the extended Euclidean algorithm in $F[t]$). Otherwise, $F[t]/(f)$ contains nonzero zero divisors — including $(f_1 \bmod f)$ and $(f_2 \bmod f)$ for any polynomials f_1 and f_2 of positive degree such that $f = f_1 f_2$.

Hence we can conclude that a semi-simple algebra $A = B_1 \oplus B_2 \oplus \cdots \oplus B_k$ with simple components $B_1, B_2, \ldots, B_k$ is a simple algebra (and $k = 1$) if we can find an element a of Centre(A) with $F[a] = \text{Centre}(A)$, such that the minimal polynomial of a is irreducible in $F[t]$.

Elements a of Centre(A) whose minimal polynomials are reducible in $F[t]$ are also useful. Suppose now that

$$a = \beta_1 + \beta_2 + \cdots + \beta_k \in A,$$

with $\beta_i \in \text{Centre}(B_i)$ for $1 \le i \le k$. Suppose f is the minimal polynomial of a over F, and that h_i is the minimal polynomial of β_i over F, for $1 \le i \le k$. It follows by Proposition 2.4.6 that the polynomials $h_1, h_2, \ldots, h_k$ are all irreducible in $F[t]$. It is also clear that f is the least common multiple of $h_1, h_2, \ldots, h_k$. Hence f is squarefree in $F[t]$, and has factorisation

$$f = f_1 f_2 \cdots f_l$$

for some $l > 0$ and for distinct polynomials $f_1, f_2, \ldots, f_l \in \{ h_1, h_2, \ldots, h_k \}$ (thus $f_1, f_2, \ldots, f_l$ are distinct, while $h_1, h_2, \ldots, h_k$ need not be). Since $f_1, f_2, \ldots, f_l$ are pairwise relatively prime, there exist polynomials $g_1, g_2, \ldots, g_l$ in $F[t]$, each with degree less than the degree of $f_1 f_2 \cdots f_l = f$, such that

$$g_i \equiv 1 \pmod{f_i} \quad \text{and} \quad g_i \equiv 0 \pmod{f_j} \quad \text{for } 1 \le i,\, j \le l,\ i \ne j.$$

Let $\hat{e}_i = g_i(a) \in A$; then the elements $\hat{e}_1, \hat{e}_2, \ldots, \hat{e}_l$ can be used to obtain a partial decomposition of A, as indicated by the following facts.

Proposition 2.4.7. Let $a = \beta_1 + \beta_2 + \cdots + \beta_k \in \text{Centre}(A)$, with $\beta_i \in \text{Centre}(B_i)$ for $1 \leq i \leq k$, and let $\hat{e}_1, \hat{e}_2, \ldots, \hat{e}_l$ be as above. Then

(i) The elements $\hat{e}_1, \hat{e}_2, \ldots, \hat{e}_l$ are central idempotents in A, such that $\hat{e}_i\hat{e}_j = \hat{e}_i$ if $i = j$ and $\hat{e}_i\hat{e}_j = 0$ if $i \neq j$, for $1 \leq i, j \leq l$.

(ii) $\hat{e}_1 + \hat{e}_2 + \cdots + \hat{e}_l = 1$.

(iii) If β_{j_1} and β_{j_2} are terms in the sum $a = \beta_1 + \beta_2 + \cdots + \beta_k$ with minimal polynomials h_{j_1} and h_{j_2}, then if $h_{j_1} \neq h_{j_2}$ then there exists some i with $1 \leq i \leq l$ such that $\hat{e}_i\beta_{j_1} = \beta_{j_1}$ and $\hat{e}_i\beta_{j_2} = 0$. If $h_{j_1} = h_{j_2}$ then, for each i such that $1 \leq i \leq l$, either $\hat{e}_i\beta_{j_1} = \beta_{j_1}$ and $\hat{e}_i\beta_{j_2} = \beta_{j_2}$, or $\hat{e}_i\beta_{j_1} = \hat{e}_i\beta_{j_2} = 0$.

Proof. Let polynomials $f, f_1, f_2, \ldots, f_l$ and $g_1, g_2, \ldots, g_l$ be as given above. Clearly $\hat{e}_i = g_i(a) = g_i(\beta_1) + g_i(\beta_2) + \cdots + g_i(\beta_k)$, for $1 \leq i \leq l$. Suppose β_j has minimal polynomial $h_j = f_i$; then, since $g_i \equiv 1 \pmod{f_i}$ and $f_i(\beta_j) = 0$, it is clear that $g_i(\beta_j) = e_j$, the multiplicative identity in B_j (using $\beta_j^0 = e_j \in B_j$). Otherwise, β_j has minimal polynomial $h_j = f_s$ for some s such that $1 \leq s \leq l$ and $s \neq i$; since $g_i \equiv 0 \pmod{f_s}$ and $f_s(\beta_j) = 0$, it is clear that $g_i(\beta_j) = 0$ in this case. It follows that $\hat{e}_i$ is an idempotent in A. Since f_i is the minimal polynomial of at least one β_j, this idempotent is nonzero.

It is also clear that for each j with $1 \leq j \leq l$, $g_i(\beta_j)$ is nonzero for exactly one polynomial g_i, $1 \leq i \leq l$. Parts (i) and (ii) follow immediately. Part (iii) is also a straightforward consequence of the definition of the polynomials $g_1, g_2, \ldots, g_h$. ■

The polynomials $g_1, g_2, \ldots, g_l$ can be computed from $f_1, f_2, \ldots, f_l$ using the Chinese remainder algorithm. Since f is the minimal polynomial of an element a of A, and A is isomorphic to a subring of $M_{n\times n}(F)$ for n the dimension of A over F, it is clear that f has degree at most n. Hence the polynomials $f_1, f_2, \ldots, f_l$ can be computed from a by computing and factoring the minimal polynomial of a. The polynomials $g_1, g_2, \ldots, g_l$, and the idempotents $\hat{e}_1, \hat{e}_2, \ldots, \hat{e}_l$ can then be computed using the Chinese remainder algorithm. This last step can be performed using $n^{O(1)}$ operations in F, or using arithmetic-Boolean circuits over F, of depth $O(\log^2(n))$ and size $n^{O(1)}$ (See von zur Gathen [53] for details).

We will use this process of extracting idempotents from a single element of A as a component in algorithms to be presented later in this section. We state this process formally in the algorithm on the following page.

Proposition 2.4.8. The algorithm "Extraction of Idempotents" can be used to compute the central idempotents generated by an element a of the centre of a finite-dimensional semi-simple associative algebra $A \subseteq M_{m\times m}(F)$ of dimension n over a field F, using arithmetic-Boolean circuits over F with oracles for factorisation of squarefree polynomials in $F[t]$, of depth $O(\log^2(mn))$ and size $(mn)^{O(1)}$, plus the cost of factoring a squarefree polynomial of degree at most n in $F[t]$.

Note that the central idempotents generated by an element a of Centre(A) are not necessarily central primitive. In the worst case, $a = 1 \in A$, and the only central idempotent generated from a (by the above method) is 1 itself. In order to compute a set of simple components for A, by generating a set of central primitive idempotents, we must either find a single element "a" of Centre(A) which generates a complete set of *central primitive* idempotents, or find a way to combine the (generally imprimitive) idempotents obtained from several elements in order to obtain primitive idempotents. We discuss several approaches for solving this last part of the problem in Sections 2.4.2–2.4.4.

Algorithm **Extraction of Idempotents**

Input.
- Integer $m > 0$.
- Matrix $a \in M_{m\times m}(G)$, a member of the centre of some finite-dimensional semi-simple associative algebra A over a field G.

Output.
- Integer $l > 0$.
- Matrices $\hat{e}_1, \hat{e}_2, \ldots, \hat{e}_l \in G[a]$ such that each matrix $\hat{e}_i$ is a nonzero idempotent in $G[a]$ with $\hat{e}_i\hat{e}_j = \delta_{i\,j}\hat{e}_i$ for $1 \leq i,\ j \leq l$, and $\hat{e}_1 + \hat{e}_2 + \cdots + \hat{e}_l$ is the multiplicative identity in $G[a]$.

```
(1) Compute the minimal polynomial $f$ of $a$ over $G$.
(2) Compute the factorisation $f = f_1 f_2 \cdots f_l$ of $f$ in $G[t]$.
(3) For $1 \leq i \leq l$, use the Chinese remainder algorithm to compute the
    polynomial $g_i \in G[t]$ with degree less than that of $f$, such that
        $g_i \equiv 1 \pmod{f_i}$   and   $g_i \equiv 0 \pmod{f_j}$
    for $1 \leq j \leq l$, $i \neq j$.
(4) Return the integer $l$ and the idempotents $\hat{e}_i = g_i(a)$, $1 \leq i \leq l$.
```

2.4.2. The Algorithm of Friedl and Rónyai

As we noted at the end of Section 2.4.1, if $a \in \text{Centre}(A)$, then in general we only obtain a partial decomposition of A by computing the idempotents $\hat{e}_1, \hat{e}_2, \ldots, \hat{e}_l$ obtained from a using the algorithm "Extraction of Idempotents". In the extreme case (for example, when $a = 1$), we do not decompose A at all. We will show later that for some fields F there exist algebras A such that no single element a of A can be used to decompose A completely by the above method. Hence, (for some fields) we must consider several elements of the algebra A in order to decompose A completely.

As we noted in Section 2.4.1, the simple components of A are easily computed from the simple components of the commutative subalgebra Centre(A). Henceforth, we will assume that A is commutative (in order to simplify the presentation, without weakening the results).

Friedl and Rónyai ([43]) solve this problem by processing each of the elements $a_1, a_2, \ldots, a_n$ of a basis for a (commutative) matrix algebra A. They maintain a list of finite algebraic extensions of F, each of the form $F[a]$ for some $a \in A$, which are the simple components of the smallest semi-simple algebra over F containing the elements seen so far. Initially, this list consists of the single field F. After all the elements of a basis have been processed, it consists of the simple components of A.

Suppose now that Friedl and Rónyai's algorithm has been used to process the first i elements, $a_1, a_2, \ldots, a_i$, of a basis for A, and that the components

$$C_1 = F[b_1], \quad C_2 = F[b_2], \quad \cdots \quad C_h = F[b_h]$$

with identity elements $\bar{e}_1, \bar{e}_2, \ldots, \bar{e}_h$ respectively, have been obtained. (Then $\bar{e}_j \in C_j$ for $1 \leq j \leq h$, and $\bar{e}_1 + \bar{e}_2 + \cdots \bar{e}_h = 1$). In order to process the next element a_{i+1} of the basis, the minimal polynomial of the element $a_{i+1,\,j} = a_{i+1}\bar{e}_j$ is computed over the field extension C_j of F, for $1 \leq j \leq h$. If the minimal polynomial of $a_{i+1,\,j}$ over C_j is irreducible in $C_j[t]$, then the element $a_{i+1,\,j}$ is adjoined to C_j, to obtain a larger field $\hat{C}_j$ which replaces C_j in the list of components. A primitive element $\hat{b}_j$ of $\hat{C}_j$ is also computed, so that arithmetic can be performed over the field extension $\hat{C}_j$ of F in later steps. If the minimal polynomial of $a_{i+1,\,j}$ over C_j is reducible in $C_j[t]$, then $a_{i+1,\,j}$ is used to generate a set of idempotents in $C_j[a_{i+1,\,j}]$ using the algorithm "Extraction of Idempotents", *performing arithmetic over the field $G = C_j$*. These idempotents are used as the identity elements of a set of components $\hat{C}_{j\,1}, \hat{C}_{j\,2}, \ldots, \hat{C}_{j\,r}$ which replace C_j in the list of components.

The algorithm is stated in more detail by Friedl and Rónyai [43]. We use it to decompose an algebra in the following example.

Example 2.4.9. Let $F = \mathbb{F}_2$, and consider the matrix algebra $A \cong \mathbb{F}_4 \oplus \mathbb{F}_{16}$, which is contained in $M_{6\times 6}(F)$ and generated as a ring over F by the elements $(\alpha, 0)$ and $(0, \beta)$ (for $\alpha \in \mathbb{F}_4$ with minimal polynomial $t^2 + t + 1$ over $\mathbb{F}_2$ and $\beta \in \mathbb{F}_{16}$ with minimal polynomial $t^4 + t^3 + t^2 + t + 1$ over $\mathbb{F}_2$), with

$$(\alpha, 0) = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \qquad (0, \beta) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}.$$

Suppose we are given the following basis for A (with componentwise addition and multiplication, for ordered pairs).

$$\begin{array}{ll} a_1 = (1, 1) & a_5 = (\alpha, \beta^2) \\ a_2 = (\alpha, 1 + \beta^2 + \beta^3) & a_6 = (1, \beta) \\ a_3 = (\alpha, 1) & a_7 = (1, \beta^2) \\ a_4 = (\alpha, \beta) & a_8 = (1, \beta^3). \end{array}$$

It is easily checked that this is a basis for an algebra $A \cong \mathbb{F}_4 \oplus \mathbb{F}_{16}$ over $F = \mathbb{F}_2$.

We begin by computing and factoring the minimal polynomial (over F) of

$$a_1 = (1, 1) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Clearly the minimal polynomial, $t + 1$, is irreducible in $\mathbb{F}_2[t]$. Since it is also linear, we obtain a single component $C_{1\,1} = F[a_1] \cong \mathbb{F}_2$, with identity element $e_{1\,1} = a_1$ and generator $b_{1\,1} = a_1$.

We next consider the element

$$a_2 = (\alpha, 1 + \beta^2 + \beta^3) = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}.$$

The minimal polynomial of $e_{1\,1}a_2$ over $C_{1\,1}$ is $t^2 + t + 1$, an irreducible polynomial in $C_{1\,1}[t]$. We adjoin this to $C_{1\,1}$ to obtain a single component $C_{2\,1} \cong \mathbb{F}_4$, with identity element $e_{2\,1} = a_1$ and generator $b_{2\,1} = e_{1\,1}a_2 = a_2$, for the algebra generated by a_1 and a_2.

The third element of our basis is

$$a_3 = (\alpha, 1) = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

The minimal polynomial of $e_{2\,1}a_3$ over $C_{2\,1}$ is

$$t^2 + (a_2 + 1)t + a_2 = (t + a_2)(t + 1).$$

This has factors

$$f_1 = (t + a_2) \qquad \text{and} \qquad f_2 = (t + 1)$$

in $C_{2\,1}$. We use these to generate idempotents $e_{3\,1}$ and $e_{3\,2}$ in the algebra generated by a_1, a_2, and a_3 by computing polynomials g_1 and g_2 in $C_{2\,1}[t]$ such that

$$\begin{aligned} g_1 &\equiv 1 \pmod{f_1}, & g_2 &\equiv 0 \pmod{f_1}, \\ g_1 &\equiv 0 \pmod{f_2}, & g_2 &\equiv 1 \pmod{f_2}. \end{aligned}$$

Using the Chinese remainder algorithm over $C_{2\,1}[t]$, we obtain polynomials

$$\begin{aligned} g_1 &= a_2t + a_2 = a_2(t + 1) = a_2(t + a_2) + 1, \\ g_2 &= a_2t + a_2 + 1 = a_2(t + 1) + 1 = a_2(t + a_2), \end{aligned}$$

which we evaluate at a_3 to obtain the idempotents

$$e_{3\,1} = (1, 0) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \qquad e_{3\,2} = (0, 1) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

We use these to split $C_{2\,1}$ into two components, $C_{3\,1}$, with identity element $e_{3\,1}$ and generator $b_{3\,1} = e_{3\,1}b_{2\,1} = e_{3\,1}a_2$, and $C_{3\,2}$, with identity element $e_{3\,2}$ and generator $b_{3\,2} = e_{3\,2}b_{2\,1} = e_{3\,2}a_2$. The minimal polynomial of $e_{3\,i}a_3$ over $C_{3\,i}$ is linear, for $i \in \{1, 2\}$; hence we cannot extend either of these components further using a_3. Thus we have two components $C_{3\,1}$ and $C_{3\,2}$ of the algebra generated by a_1, a_2, and a_3, with each component isomorphic to $\mathbb{F}_4$.

We next consider the element

$$a_4 = (\alpha, \beta) = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}.$$

The matrix $e_{3\,1}a_4$ has minimal polynomial $t + b_{3\,1}$ over $C_{3\,1}$. Since this is linear, we neither split $C_{3\,1}$ nor extend this component to obtain a larger field; we set $C_{4\,1} = C_{3\,1}$. On the other hand, the matrix $e_{3\,2}a_4$ has minimal polynomial

$$t^2 + t + 1 + b_{3\,2}$$

over $C_{3\,2}$. Since this is irreducible in $C_{3\,2}[t]$, we obtain a larger component $C_{4\,2} = C_{3\,2}[e_{3\,2}a_4]$. Computing a primitive generator, we see that $C_{4\,2}$ has identity element $e_{4\,2} = e_{3\,2}$ and generator $b_{4\,2} = e_{3\,2}a_4$. Hence the algebra generated by a_1, a_2, a_3, and a_4 over F has two components, $C_{4\,1}$ (isomorphic to $\mathbb{F}_4$) and $C_{4\,2}$ (isomorphic to $\mathbb{F}_{16}$).

We consider the element a_i, for $5 \le i \le 8$, by computing the minimal polynomial of $e_{4\,j}a_i$ over $C_{4\,j}$, for $j = 1, 2$. In each case, we find that this minimal polynomial is linear. Hence we do not change the components $C_{4\,1}$ and $C_{4\,2}$. We conclude that

$$A = B_1 \oplus B_2,$$

for simple components

$$B_1 = C_{4\,1} \cong \mathbb{F}_4 \qquad \text{and} \qquad B_2 = C_{4\,2} \cong \mathbb{F}_{16}.$$

Friedl and Rónyai show that this algorithm can be used to extract the simple components of a finite-dimensional semi-simple associative algebra over a number field or over a finite field using a polynomial number of Boolean operations.

Theorem 2.4.10. (Friedl and Rónyai [43]). Suppose $A \subseteq M_{m\times m}(F)$ is a finite-dimensional semi-simple associative algebra of dimension n over a field F.

(i) If F is a number field, then bases for the simple components of A can be computed from a basis for A using $N^{O(1)}$ Boolean operations, for input size N.

(ii) If $F = \mathbb{F}_{p^l}$, then bases for the simple components of A can be computed from a basis for A, using $(nmpl)^{O(1)}$ Boolean operations, or using a probabilistic Boolean algorithm using $(nml \log p)^{O(1)}$ Boolean operations (that is, $N^{O(1)}$ operations for input size N), which either successfully performs the above computation, or indicates "failure", failing with probability at most $1/2$.

Applying the results about arithmetic over field extensions discussed in Section 1 (in particular, Theorem 1.3.1(i) and Theorem 1.4.15), we also obtain the following reduction from Friedl and Rónyai's algorithm.

Theorem 2.4.11. Suppose $A \subseteq M_{m\times m}(F)$ is a finite-dimensional semi-simple associative algebra of dimension n over a perfect field F; then bases for the simple components of A can be computed from a basis for A using arithmetic-Boolean circuits over F (with oracles for factorisation of squarefree polynomials in $F[t]$) with size $(nm)^{O(1)}$ (that is, size $N^{O(1)}$, for input size N).

2.4.3. A New Algorithm for Simple Components

We now present an new algorithm for "Extraction of Simple Components" for finite-dimensional semi-simple algebras, which is correct for arbitrary fields of characteristic zero, as well as arbitrary finite fields $\mathbb{F}_{p^l}$. Again, we simplify the presentation by assuming A is commutative (and applying the results of Section 2.4.1 to obtain an algorithm which is correct for arbitrary finite-dimensional semi-simple algebras). Instead of considering elements of a basis for an algebra A in sequence, to refine a single decomposition of A, we use these elements independently, to obtain several different partial decompositions. We then combine these decompositions to obtain a single, complete decomposition of A into simple components.

Once again, let A be a commutative, finite-dimensional, semi-simple associative algebra over F with simple components $B_1, B_2, \ldots, B_k$ over F, and let $a_1, a_2, \ldots, a_n$ be a basis for A over F. Let $a = \beta_1 + \beta_2 + \cdots + \beta_k \in A$, with $\beta_i \in B_i$ for $1 \leq i \leq k$. Let $1 \leq i, j \leq k$, with $i \neq j$; we say that *a splits* components B_i and B_j if β_i and β_j have distinct minimal polynomials over F — so that the element a can be used to generate an idempotent e of A (using the algorithm "Extraction of Idempotents", with computations over the ground field F) such that $eB_i = B_i$ and $eB_j = (0)$. We will show that B_i and B_j are split by some element a_h of our basis for A, for each pair of components B_i and B_j with $i \neq j$.

We prove this by considering traces of elements over field extensions. Suppose now that $E = F[t]/(f)$, for f monic and irreducible of degree l in $F[t]$, and for F perfect. Let $\alpha_1, \alpha_2, \ldots, \alpha_l$ be the roots of f in an algebraic closure H of F; since f is separable, these roots are distinct. The field $E = F[t]/(f)$ is isomorphic to the field $F[\alpha_1]$; we use the isomorphism taking (t mod f) to α_1 to embed E in the closure H.

Definition 2.4.12. Let $\gamma = c_0 + c_1\alpha_1 + c_2\alpha_1^2 + \cdots + c_{l-1}\alpha_1^{l-1} \in F[\alpha_1]$, for $c_0, c_1, \ldots, c_{l-1} \in F$. Since α_1 has minimal polynomial f of degree l over F, the coefficients $c_0, c_1, \ldots, c_{l-1}$ are unique (for γ). The *trace* of γ over F, $T_{F[\alpha_1]/F}(\gamma)$, is

$$T_{F[\alpha_1]/F}(\gamma) = \sum_{i=1}^{l}(c_0 + c_1\alpha_i + c_2\alpha_i^2 + \cdots + c_{l-1}\alpha_i^{l-1}).$$

Since $T_{F[\alpha_1]/F}(\gamma)$ is fixed by the Galois group of H (a splitting field of f) over F, it is clear that $T_{F[\alpha_1]/F}(\gamma) \in F$.

Like the norm over F, $N_{F[\alpha_1]/F}$ (defined in Section 1.4), $T_{F[\alpha_1]/F}$ is a map from $F[\alpha_1]$ to F. Unlike the norm, it is F-linear; if $\alpha, \beta \in F[\alpha_1]$ and $c \in F$, then

$$T_{F[\alpha_1]/F}(\alpha + c\beta) = T_{F[\alpha_1]/F}(\alpha) + cT_{F[\alpha_1]/F}(\beta).$$

Lemma 2.4.13. Let $E = F[\alpha]$ be a finite algebraic extension of a perfect field F; then there exists an element λ of E such that $T_{E/F}(\lambda) \neq 0$.

Proof. Suppose α has minimal polynomial $f \in F[t]$ with degree l, and roots $\alpha = \alpha_1, \alpha_2, \ldots, \alpha_l$ in some algebraic closure of E. Since F is perfect, f is a separable polynomial, and the roots $\alpha_1, \alpha_2, \ldots, \alpha_l$ are distinct. Suppose $T_{E/F}(\lambda) = 0$ for all $\lambda \in E$; then, in particular,

$$T_{E/F}(1) = T_{E/F}(\alpha) = T_{E/F}(\alpha^2) = \cdots = T_{E/F}(\alpha^{l-1}) = 0.$$

Since $T_{E/F}(\alpha^i) = \alpha_1^i + \alpha_2^i + \cdots + \alpha_l^i$ for $0 \leq i \leq l$, this is equivalent to the statement

$$\begin{bmatrix} 1 & 1 & \cdots & 1 \\ \alpha_1 & \alpha_2 & \cdots & \alpha_l \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_1^{l-1} & \alpha_2^{l-1} & \cdots & \alpha_l^{l-1} \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

The coefficient matrix of this system is the *Vandermonde matrix* for $\alpha_1, \alpha_2, \ldots, \alpha_l$. The formula for the determinant of this matrix is well known; the determinant is

$$\prod_{i=2}^{l} \prod_{j=1}^{i-1} (\alpha_i - \alpha_j) \neq 0,$$

contradicting the fact that we have a nonzero element of the nullspace of this matrix. We conclude that there exists some $\lambda \in E$ (in particular, $\lambda \in \{ 1, \alpha, \alpha^2, \ldots, \alpha^{l-1} \}$) such that $T_{E/F}(\lambda) \neq 0$. ■

Lemma 2.4.14. Let $\beta_1, \beta_2 \in E$, for E a finite algebraic extension of a perfect field F, such that β_1 and β_2 have the same minimal polynomial over F. Then $T_{E/F}(\beta_1) = T_{E/F}(\beta_2)$.

Proof. If $\beta_1 = \beta_2$ then the claim is trivial; we now suppose $\beta_1 \neq \beta_2$.

Let $F_1 = F[\beta_1]$, $F_2 = F[\beta_2]$, and let $\beta_1, \beta_2, \ldots, \beta_l$ be the conjugates of β over F. Then the trace of β_1 in F_1 (over F) and the trace of β_2 in F_2 (over F) both equal $\beta_1 + \beta_2 + \cdots + \beta_l$.

Now $F \subseteq F_1 \subseteq E$, $F \subseteq F_2 \subseteq E$, and F_1 and F_2 are isomorphic (as field extensions of F). Thus $[E : F_1] = [E : F_2] = k$, for some $k > 0$. It follows that the trace of β_1 in E (over F) is k times the trace of β_1 in F_1 (over F), while the trace of β_2 in E (over F) is k times the trace of β_2 in F_2 (again, over F) (see van der Waerden [117], Section 6.11 for a proof). Thus $T_{E/F}(\beta_1) = k \cdot T_{F[\beta_1]/F}(\beta_1) = k \cdot (\beta_1 + \beta_2 + \cdots + \beta_l) = k \cdot T_{F[\beta_2]/F}(\beta_2) = T_{E/F}(\beta_2)$, as claimed. ■

Theorem 2.4.15. Let a_1, a_2, …, a_n be a basis over F for a commutative finite-dimensional semi-simple associative algebra A over F, with simple components B_1, B_2, …, B_k. Suppose the field F satisfies one or more of the following properties.

(i) F has characteristic zero;
(ii) F is a finite field;
(iii) F is algebraically closed.

Then if $1 \leq i$, $j \leq k$ and $i \neq j$ then there exists some element a_h of the above basis for A such that a_h splits B_i and B_j (as defined on page 95).

Proof. Suppose, to the contrary, that no element of the basis splits the components B_i and B_j. We write

$$\begin{aligned} a_1 &= \beta_{1\,1} + \beta_{1\,2} + \cdots + \beta_{1\,k} \\ a_2 &= \beta_{2\,1} + \beta_{2\,2} + \cdots + \beta_{2\,k} \\ &\vdots \\ a_n &= \beta_{n\,1} + \beta_{n\,2} + \cdots + \beta_{n\,k} \end{aligned}$$

with $a_{r\,s} \in B_s$ for $1 \leq r \leq n$ and $1 \leq s \leq k$. Since a_r does not split B_i and B_j, $\beta_{r\,i}$ and $\beta_{r\,j}$ must have the same minimal polynomial over F, for $1 \leq r \leq n$. (Clearly $\beta_{r\,i}$ and $\beta_{r\,j}$ are algebraic over F, since they belong to a finite-dimensional semi-simple associative algebra over F.) We consider fields F satisfying the conditions (i), (ii), and (iii) stated in the theorem separately.

(i) Since a_1, a_2, …, a_n is a basis for A over F, there exist elements γ_1, γ_2, …, γ_n of F such that

$$\gamma_1 a_1 + \gamma_2 a_2 + \cdots + \gamma_n a_n = e_i,$$

for e_i the identity element of B_i. It follows that

$$\gamma_1 \beta_{i\,1} + \gamma_2 \beta_{i\,2} + \cdots + \gamma_n \beta_{i\,n} = 1,$$

and

$$\gamma_1 \beta_{j\,1} + \gamma_2 \beta_{j\,2} + \cdots + \gamma_n \beta_{j\,n} = 0.$$

Let E be a finite extension of F which contains the elements $\beta_{i\,r}$ and $\beta_{j\,r}$ for $1 \leq r \leq n$. (That is, let E be an extension field of F with subfields isomorphic, as extensions of F, to B_1 and B_2, respectively.) We consider the trace (in E over F) of elements of E. Since $\beta_{i\,r}$ and $\beta_{j\,r}$ are conjugates over F, $T_{E/F}(\beta_{i\,r}) = T_{E/F}(\beta_{j\,r})$, for $1 \leq r \leq n$, by Lemma 2.4.14.

Since the trace is an F-linear map, it follows that

$$\begin{aligned}[E:F] &= T_{E/F}(1) \\ &= T_{E/F}(\gamma_1\beta_{i\,1} + \gamma_2\beta_{i\,2} + \cdots + \gamma_n\beta_{i\,n}) \\ &= \gamma_1 T_{E/F}(\beta_{i\,1}) + \gamma_2 T_{E/F}(\beta_{i\,2}) + \cdots + \gamma_n T_{E/F}(\beta_{i\,n}) \\ &= \gamma_1 T_{E/F}(\beta_{j\,1}) + \gamma_2 T_{E/F}(\beta_{j\,2}) + \cdots + \gamma_n T_{E/F}(\beta_{j\,n}) \\ &= T_{E/F}(\gamma_1\beta_{j\,1} + \gamma_2\beta_{j\,2} + \cdots + \gamma_n\beta_{j\,n}) \\ &= T_{E/F}(0) = 0,\end{aligned}$$

contradicting that fact that the field F has characteristic zero, and $[E:F] \geq 1$.

(ii) Suppose $F = \mathbb{F}_{p^l}$. Now the components B_i and B_j are both fields, with $B_i = \mathbb{F}_{p^l}[\beta_{1\,i}, \beta_{2\,i}, \ldots, \beta_{n\,i}]$, and $B_j = \mathbb{F}_{p^l}[\beta_{1\,j}, \beta_{2\,j}, \ldots, \beta_{n\,j}]$. We consider both B_i and B_j to be embedded in some larger extension of $\mathbb{F}_{p^l}$. Now B_i and B_j are both *normal fields:* if γ is an element of B_i, E is an extension of B_i (so $\mathbb{F}_{p^l} \subseteq B_i \subseteq E$), and $\hat{\gamma} \in E$ such that γ and $\hat{\gamma}$ have the same minimal polynomial over $\mathbb{F}_{p^l}$, then $\hat{\gamma} \in B_i$ as well. In particular, taking as E the smallest field containing both B_i and B_j, we see that $\beta_{r\,j} \in B_i$ for $1 \leq r \leq n$, since $\beta_{r\,i} \in B_i$ and $\beta_{r\,j}$ and $\beta_{r\,i}$ have the same minimal polynomial over F. Since B_i and B_j are both fields, and B_j is generated over $\mathbb{F}_{p^l}$ by the elements $\beta_{r\,j}$, it follows that $B_j \subseteq B_i$. Clearly, $B_i \subseteq B_j$ as well, so $B_i = B_j$ (in this embedding). It follows that the components B_i and B_j of A are isomorphic (as extensions of F), and that the trace of $\beta_{r\,i}$ in B_i (over F), $T_{B_i/F}(\beta_{r\,i})$, equals the trace of $\beta_{r\,j}$ in B_j (again, over F), $T_{B_j/F}(\beta_{r\,j}$, for $1 \leq r \leq n$.

By Lemma 2.4.13, there exists some element λ of B_i such that the trace of λ over F is nonzero. Since $a_1, a_2, \ldots, a_n$ is a basis for A over F, it is clear that there exist elements $\gamma_1, \gamma_2, \ldots, \gamma_n$ of F such that

$$\gamma_1\beta_{1\,i} + \gamma_2\beta_{2\,i} + \cdots + \gamma_n\beta_{n\,i} = \lambda,$$

and

$$\gamma_1\beta_{1\,j} + \gamma_2\beta_{2\,j} + \cdots + \gamma_n\beta_{n\,j} = 0.$$

Now, since $T_{B_i/F}(\beta_{r\,i}) = T_{B_j/F}(\beta_{r\,j})$ for $1 \leq r \leq n$, we conclude (as in (i)) that $T_{B_i/F}(\lambda) = T_{B_j/F}(\lambda) = 0$, contradicting the choice of λ.

(iii) If F is algebraically closed then, since $\beta_{r\,i}$ and $\beta_{r\,j}$ are conjugates over F, $\beta_{r\,i} = \beta_{r\,j}$ for $1 \leq r \leq n$, again contradicting the fact that there must exist constants $\gamma_1, \gamma_2, \ldots, \gamma_n \in F$ such that

$$\lambda_1\beta_{1\,i} + \lambda_2\beta_{2\,i} + \cdots + \lambda_n\beta_{n\,i} = 1,$$

and

$$\lambda_1\beta_{1\,j} + \lambda_2\beta_{2\,j} + \cdots + \lambda_n\beta_{n\,j} = 0. \quad \blacksquare$$

Now suppose we have two sets of central idempotents in the algebra A, $e_{1\,1}, e_{1\,2}, \ldots, e_{1\,k_1}$ and $e_{2\,1}, e_{2\,2}, \ldots, e_{2\,k_2}$, for $k_1, k_2 > 0$, with

$$e_{1\,1} + e_{1\,2} + \cdots + e_{1\,k_1} = 1 = e_{2\,1} + e_{2\,2} + \cdots + e_{2\,k_2},$$

and with $e_{1\,i}e_{1\,j} = \delta_{i\,j}e_{1\,i}$ for $1 \leq i, j \leq k_1$, and $e_{2\,i}e_{2\,j} = \delta_{i\,j}e_{2\,i}$ for $1 \leq i, j \leq k_2$. We obtain a third set of central idempotents, $e_{3\,1}, e_{3\,2}, \ldots, e_{3\,k_3}$, which forms a *refinement* of the above sets of idempotents, by forming the set of products

$$e_{1\,i}e_{2\,j}, \qquad \text{for } 1 \leq i \leq k_1, \text{ and } 1 \leq j \leq k_2,$$

and discarding all products which equal 0. Now if B_i and B_j are simple components which are separated by some idempotent $e_{1\,r}$, so $e_{1\,r}B_i = B_i$ and $e_{1\,r}B_j = (0)$, then there is an idempotent $e_{3\,s}$ which also separates B_i and B_j in this way. Similarly, if B_i and B_j are separated by some idempotent $e_{2\,r}$, then there is an idempotent $e_{3\,s}$ separating B_i and B_j as well. If F is a field of characteristic zero, a finite field, or algebraically closed, then it follows that we can compute the primitive idempotents of a finite-dimensional semi-simple associative algebra A over F by computing a set of idempotents for each element of a basis for the centre of A, and then computing a common refinement of these sets of idempotents.

It is clear that the idempotents $e_{3\,1}, e_{3\,2}, \ldots, e_{3\,k_3}$ which refine the sets of idempotents $e_{1\,1}, e_{1\,2}, \ldots, e_{1\,k_1}$ and $e_{2\,1}, e_{2\,2}, \ldots, e_{2\,k_2}$ can be computed using a polynomial number of operations over the ground field, F. We now consider the cost of computing these idempotents in parallel. If $A \subseteq M_{m\times m}(F)$ then it is clear that each product $e_{1\,i}e_{2\,j}$ can be computed using arithmetic-Boolean circuits of depth $O(\log m)$ and size $m^{O(1)}$; we can also decide whether each result is nonzero at this cost. Since we can count the number of nonzero results, and select the i^{th} nonzero result from the list

$$e_{1\,1}e_{2\,1}, e_{1\,1}e_{2\,2}, \ldots, e_{1\,1}e_{2\,k_2}, e_{1\,2}e_{2\,1}, \ldots, e_{1\,k_1}e_{2\,k_2},$$

(for $i \leq m^2$) at this cost as well, it follows that we can compute the new set of idempotents $e_{3\,1}, e_{3\,2}, \ldots, e_{3\,k_3}$ using arithmetic-Boolean circuits of depth $O(\log m)$ and size polynomial in m.

If we are given l sets of idempotents $e_{i\,1}, e_{i\,2}, \ldots, e_{i\,k_i}$, for $1 \leq i \leq l$, then we can compute a common refinement $e_1, e_2, \ldots, e_k$ of these idempotents by treating sets in pairs. The algorithm "Refinement of Idempotents" on the following page uses a divide-and-conquer approach to solve this problem. This is used in the algorithm "Simple Components via Idempotents of Basis", on the next page.

Algorithm **Refinement of Idempotents**

Input.
- Integers m, $n > 0$.
- Integer $k_i > 0$, and idempotents $e_{i\,1}, e_{i\,2}, \ldots, e_{i\,k_i}$ in $M_{m\times m}(F)$, for $1 \le i \le n$, such that $e_{i\,1} + e_{i\,2} + \cdots + e_{i\,k_i} = 1$ and $e_{i\,r}e_{i\,s} = \delta_{r\,s}e_{i\,r}$ for $1 \le r,\, s \le k_i$.

Output.
- Integer $k > 0$ and idempotents $e_1, e_2, \ldots, e_k$ of $M_{m\times m}(F)$ such that $e_1 + e_2 + \cdots + e_k = 1$, $e_re_s = \delta_{r\,s}e_r$, for $1 \le r,\, s \le k$, such that each idempotent $e_{i\,r}$ is the sum of some subset of the idempotents $e_1, e_2, \ldots, e_k$, for $1 \le i \le n$ and $1 \le r \le k_i$, and such that each idempotent e_i is the product of some subset of the idempotents given as input.

(1) If $n = 1$ then return the integer $k = k_1$ and the idempotents $e_{1\,1}, e_{1\,2}, \ldots, e_{1\,k_1}$. Otherwise, perform steps 2 and 3.

(2) Let $\hat{n} = \lceil \frac{n}{2} \rceil$; perform steps 2a and 2b.

(2a) Use algorithm “Refinement of Idempotents” recursively, with inputs m and $\hat{n}$, and integer k_i and idempotents $e_{i\,1}, e_{i\,2}, \ldots, e_{i\,k_i}$ for $1 \le i \le \hat{n}$, to compute an integer $\hat{k}_1$ and a set of idempotents $\hat{e}_{1\,1}, \hat{e}_{1\,2}, \ldots, \hat{e}_{1\,\hat{k}_1}$ refining these idempotents.

(2b) Use algorithm “Refinement of Idempotents” recursively, with inputs m, $n - \hat{n}$, and integer k_i and idempotents $e_{i\,1}, e_{i\,2}, \ldots, e_{i\,k_i}$ for $\hat{n} + 1 \le i \le n$, to compute an integer $\hat{k}_2$ and a set of idempotents $\hat{e}_{2\,1}, \hat{e}_{2\,2}, \ldots, \hat{e}_{2\,\hat{k}_2}$ refining these idempotents.

(3) Compute the products $\hat{e}_{1\,r}\hat{e}_{2\,s}$ for $1 \le r \le \hat{k}_1$ and $1 \le s \le \hat{k}_2$. Set k to be the number of nonzero products obtained, and set $e_1, e_2, \ldots, e_k$ to be these nonzero idempotents. Return these values.

Algorithm **Simple Components via Idempotents of Basis**

Input.
- Integers n, $m > 0$.
- Matrices $a_1, a_2, \ldots, a_n \in M_{m\times m}(F)$, which form the basis for a finite-dimensional semi-simple associative algebra $A \subseteq M_{m\times m}(F)$ of dimension n over F.

Output.
- Integer $k > 0$, the number of simple components of A.
- Integers $n_1, n_2, \ldots, n_k > 0$, with $n_1 + n_2 + \cdots + n_k = n$, such that n_i is the dimension of simple component B_i of A.
- Elements $\mu_{i\,j\,l}$ of F, for $1 \le i \le k$, $1 \le j \le n_i$, and $1 \le l \le n$, defining elements $b_{i\,j} = \sum_{l=1}^{n} \mu_{i\,j\,l} a_l$ of A such that
  - (1) $b_{i\,1}, b_{i\,2}, \ldots, b_{i\,n_i}$ is a basis for the simple component B_i over F, and hence
  - (2) $b_{1\,1}, \ldots, b_{k\,n_k}$ is a basis for A over F.
- Matrices $c_{i\,j} \in M_{n_i\times n_i}(F)$ such that $c_{i\,1}, c_{i\,2}, \ldots, c_{i\,n_i}$ is the basis for a matrix algebra isomorphic to B_i, for $1 \le i \le k$.

(1) Compute a basis $\hat{a}_1, \hat{a}_2, \ldots, \hat{a}_l$ for Centre(A) over F.

(2) For $1 \le i \le l$, use the algorithm "Extraction of Idempotents" with input $\hat{a}_i$ to compute integer $k_i > 0$ and idempotents $e_{i\,1}, e_{i\,2}, \ldots, e_{i\,k_i}$.

(3) Use the algorithm "Refinement of Idempotents" with inputs m, $l > 0$ and integer k_i and idempotents $e_{i\,1}, e_{i\,2}, \ldots, e_{i\,k_i}$, for $1 \le i \le l$, to generate an integer $k > 0$ and idempotents $e_1, e_2, \ldots, e_k$ which are common refinements of the idempotents computed in step 2.

(4) Use the algorithm "Simple Components via Central Primitive Idempotents" to compute the remaining values to be produced as output.

Theorem 2.4.16. Suppose $A \subseteq M_{m\times m}(F)$ is a finite-dimensional semi-simple associative algebra of dimension n over a field F.

(i) If F has characteristic zero, is a finite field, or is algebraically closed, then given a basis for A, bases for the simple components of A can be computed using arithmetic-Boolean circuits over F (with oracles for factorisation of squarefree polynomials in $F[t]$), with depth $O(\log^2(mn))$ and size $(mn)^{O(1)}$, plus the cost of factoring n squarefree polynomials, each with degree at most m, in parallel.

(ii) If $F = \mathbb{F}_{p^l}$, then bases for the simple components of A can be computed from a basis for A, using Boolean circuits of size $(nmpl)^{O(1)}$ and of depth $O(log^3(nmpl))$, or using a probabilistic algorithm for factorisation of polynomials over finite fields, using Boolean circuits (with extra nodes producing random bits) of size $(nml \log p)^{O(1)}$ and depth $O(\log^2(nm)\log^2 l \log p)$, which either successfully perform the above computation, or indicate "failure", failing with probability at most $1/2$.

Proof. We use the algorithm "Simple Components via Idempotents of Basis" to perform this computation. It follows from Theorem 2.4.15 that the algorithm is correct for the fields mentioned in the statement of the theorem. The upper bounds on circuit size and depth stated in (i), and for deterministic computations in (ii), follow from bounds stated for factorisation of polynomials over number fields and finite fields in Section 1.4, for linear algebra in Section 1.3, and for parallel algorithms for the "Chinese remainder" problem for polynomials, as stated by von zur Gathen [53].

As stated in Section 1.4, von zur Gathen [52] shows that the probabilistic algorithm of Cantor and Zassenhaus [19] can be used to produce Boolean circuits of size $(ml \log p)^{O(1)}$ and depth $O(\log^2 m \log^2 l \log p)$, which either successfully factor a squarefree polynomial of degree m over $\mathbb{F}_{p^l}$ (with probability at least $1/2$), or report "failure". It is clear that the probability of success can be improved to $1-1/k$, for arbitrarily large k, by executing this algorithm $\lceil \log k \rceil$ times, independently and in parallel, and then using the output of any of these trials which does not report failure, reporting failure only if all of the trials fail. In particular, we obtain circuits of depth $O(\log^2 m \log^2 l \log p + \log\log m)$ and size $(ml \log p \log n)^{O(1)}$ for this problem, which return "failure" with probability at most $1/(2n)$. Since the algorithm "Simple Components via Idempotents of Basis" requires the factorisation of exactly n polynomials, it is clear that all factorisations will succeed, with probability at least $1/2$, if these circuits are used for factorisation (with random bits chosen independently). The rest of the timing analysis stated in (ii) is straightforward. ■

Example 2.4.17. Consider the field $F = \mathbb{F}_2$, the algebra $A \cong \mathbb{F}_4 \oplus \mathbb{F}_{16}$, and the basis $a_1, a_2, \ldots, a_8$ for A over F given in Example 2.4.9. The elements $a_1, a_2, \ldots, a_8$ have minimal polynomials $f_1, f_2, \ldots, f_8$ over F respectively, for

$$\begin{aligned} f_1 &= t+1, & f_5 &= (t^2+t+1)(t^4+t^3+t^2+t+1), \\ f_2 &= t^2+t+1, & f_6 &= (t+1)(t^4+t^3+t^2+t+1), \\ f_3 &= (t+1)(t^2+t+1), & f_7 &= (t+1)(t^4+t^3+t^2+t+1), \\ f_4 &= (t^2+t+1)(t^4+t^3+t^2+t+1), & f_8 &= (t+1)(t^4+t^3+t^2+t+1). \end{aligned}$$

Since polynomials f_1 and f_2 are irreducible in $F[t]$, a_1 and a_2 each generate the (single) central idempotent $1 = (1, 1)$, the identity element in A. Each of $f_3, f_4, \ldots, f_8$ are reducible in $F[t]$, so each of $a_3, a_4, \ldots, a_8$ can be used (independently) to generate the central primitive idempotents $(1, 0)$ and $(0, 1)$ for A. Computation of bases for the simple components of A from these idempotents proceeds as in Example 2.4.9.

We should note again that Friedl and Rónyai's algorithm (discussed in the last section) can also be used to compute simple components of finite-dimensional semi-simple algebras over number fields in polynomial time, and over finite fields in polynomial time using a probabilistic algorithm, with small probability of failure, for factorisation of polynomials. With some difficulty, Friedl and Rónyai's method can be adapted (to consider elements of a basis for A by a divide-and-conquer approach, rather than sequentially) to obtain a parallel algorithm for the computation of simple components of semi-simple algebras over finite fields. The time bounds obtained by this method are slightly worse than those stated in Theorem 2.4.16 — if $F = \mathbb{F}_{p^l}$ then we obtain Boolean circuits of depth $O(\log n \log^3(nmpl))$, or of depth $O(\log n \log^2(nm) \log^2 l \log p)$ for probabilistic methods, rather than the bounds stated in Theorem 2.4.16(ii). It is probably more important that our algorithm is somewhat simpler than Friedl and Rónyai's. In particular, it eliminates the use of computations over extensions of the ground field, required by the earlier algorithm. We continue the process of simplifying the method in the next section.

2.4.4. Minimising the Use of Factorisation

Since factorisation is (apparently) the most expensive part of this computation, it is in our interest to reduce the use of factorisation in our algorithm as much as possible. With this in mind, we return to the question of whether a single element a of the algebra A can be used to generate the primitive idempotents of A (hence, to decompose A completely into simple components), using the algorithm "Extraction of Idempotents". We first note a negative result.

Theorem 2.4.18. Let $F = \mathbb{F}_{p^l}$, and let $n \in \mathbb{Z}$ with $n > p^l$. Then there exists a commutative, semi-simple associative algebra A of dimension n over F, such that no single element a of A can be used to generate the primitive idempotents of A using "Extraction of Idempotents".

Proof. Let $A = F^n$; then an arbitrary element a of A has the form $(a_1, a_2, \ldots, a_n)$, for $a_i \in F$. Suppose a can be used to generate the n primitive idempotents, $(1, 0, \ldots, 0)$, $(0, 1, 0, \ldots, 0)$, $\ldots$, $(0, \ldots, 0, 1)$ of A using "Extraction of Idempotents"; then it is clear (on examination of the algorithm) that the minimal polynomial of a over F must have n irreducible factors in $F[t]$, and degree at least n. It is also clear that $g(a) = 0$ for all $a \in A$, for

$$g = \prod_{\alpha \in F} (t - \alpha) = t^{p^l} - t,$$

since $g(a) = (g(a_1), g(a_2), \ldots, g_(a_n))$. Since $|F| = p^l < n$, g is nonzero and has degree less than n in $F[t]$, giving us a contradiction. ∎

We next show that suitable elements exist in all other cases (for perfect fields).

Lemma 2.4.19. Let F be a field, and let $n \in \mathbb{Z}$ such that $|F| \geq n > 0$. Let $k > 0$; then if $F[t]$ includes an irreducible polynomial f of degree k, then it includes at least $\lceil (n(n-1))/k \rceil$ distinct monic irreducible polynomials $\hat{f}$ of degree k, which have roots in $F[t]/(f)$.

Proof. Suppose f is a monic irreducible polynomial of degree k in $F[t]$ (the result is trivial if no such polynomial exists). Let $E = F[t]/(f)$, and let α be a root of f in E (in particular, let $\alpha = t + (f)$). Now let $a, b \in F$ with $b \neq 0$, and consider the element $\alpha_{a\,b} = a + \alpha b \in E$. Since $\alpha = b^{-1}(\alpha_{a\,b} - a) \in F[\alpha_{a\,b}] \subseteq F[\alpha]$, it is clear that $F[\alpha_{a\,b}] = E$ for all $\alpha_{a\,b}$. It follows that the minimal polynomial $f_{a\,b}$ of $\alpha_{a\,b}$ over F is a monic, irreducible polynomial of degree k in $F[t]$ with a root in E.

Now since we have (at least) $n(n-1)$ choices of a and b in F, and each polynomial $f_{a\,b}$ has at most k roots in E, it follows that there are at least $\lceil (n(n-1))/k \rceil$ monic irreducible polynomials of degree k in $F[t]$ with roots in E, as claimed. ∎

Theorem 2.4.20. Let F be a perfect field which contains at least n distinct elements, and let A be a semi-simple associative algebra of dimension n over F. Then there exists some element a in the centre of A, such that the primitive idempotents of A can be generated using the algorithm "Extraction of Idempotents" with input a.

Proof. Suppose the commutative algebra Centre(A) has dimension $m \leq n$, and has simple components $B_1, B_2, \ldots, B_k$, with dimensions $m_1, m_2, \ldots, m_k$, respectively, and with primitive idempotents $e_1, e_2, \ldots, e_k$. It is clearly sufficient to prove the existence of an element $a = \alpha_1 + \alpha_2 + \cdots + \alpha_k$, with $\alpha_i \in B_i$ for $1 \leq i \leq k$, such that the minimal polynomials $f_1, f_2, \ldots, f_k$ over F of the elements $\alpha_1, \alpha_2, \ldots, \alpha_k$ respectively, are distinct.

If $k = 1$, so $m_1 = m$, then the result is trivial: We can set $a = 1$, the identity element of A. Suppose now that $k > 1$; then, since each m_i is positive and since $m_1 + m_2 + \cdots + m_k = m \leq n$, it is clear that $m_i < n$ for all i. Since F is perfect and B_i is a finite algebraic extension (field) of F, there exists an element β_i of B_i with minimal polynomial $g_i \in F[t]$, such that $B_i = F[\beta_i]$, and g_i is irreducible with degree m_i in $F[t]$. Now $B_i \cong F[t]/(g_i)$, and by Lemma 2.4.19, there exist at least $\lceil (n(n-1))/m_i \rceil \geq n$ distinct monic irreducible polynomials in $F[t]$ with degree m_i which have roots in B_i. Let $g_{i\,1}, g_{i\,2}, \ldots, g_{i\,n} \in F[t]$ be n such polynomials. We now set f_1 to be g_1, and set f_i to be any one of the polynomials $g_{i\,1}, g_{i\,2}, \ldots, g_{i\,n}$ which is not in the set $\{ f_1, f_2, \ldots, f_{i-1} \}$, for $2 \leq i \leq k$. Clearly f_i is monic and irreducible with degree m_i in $F[t]$, f_i has a root α_i in B_i, and the polynomials $f_1, f_2, \ldots, f_k$ are distinct. Thus the element $a = \alpha_1 + \alpha_2 + \cdots + \alpha_k \in$ Centre(A) has the desired properties. ■

We extend Theorem 2.4.20 to obtain an efficient algorithm for "Extraction of Simple Components" by showing that a randomly selected element of the centre of A can be used to generate the primitive idempotents of A, with arbitrarily high probability, if the ground field F is infinite.

Suppose now that $\hat{a}_1, \hat{a}_2, \ldots, \hat{a}_l$ is a basis over F for the centre of A, and consider the element

$$\hat{a}_1 x_1 + \hat{a}_2 x_2 + \cdots + \hat{a}_l x_l$$

of $A[x_1, x_2, \ldots, x_l]$ for indeterminates $x_1, x_2, \ldots, x_l$ over F. Applying Theorem 2.4.20 (to the algebra Centre(A)), we see that if $|F| \geq l$ then there exist elements $\lambda_1, \lambda_2, \ldots, \lambda_l$ of F such that the element $\hat{a}_1\lambda_1 + \hat{a}_2\lambda_2 + \cdots + \hat{a}_l\lambda_l$ of A can be used to generate the primitive idempotents of Centre(A), and of A. Now let $\phi(a)$ denote the regular matrix representation of a in Centre(A), with respect to the basis $\hat{a}_1, \hat{a}_2, \ldots, \hat{a}_l$. Extending ϕ to (Centre(A))$[x_1, x_2, \ldots, x_l]$, we obtain a matrix

$$\phi(\hat{a}_1 x_1 + \hat{a}_2 x_2 + \cdots + \hat{a}_l x_l) \in M_{l \times l}(F[x_1, x_2, \ldots, x_l]),$$

such that each entry of this matrix is linear in the indeterminates $x_1, x_2, \ldots, x_l$. The characteristic polynomial of this matrix is a polynomial with degree l in a "new" indeterminate, t, whose coefficients in t each have total degree at most l in the indeterminates $x_1, x_2, \ldots, x_l$; we will call this polynomial $\chi(x_1, x_2, \ldots, x_l, t)$. We obtain the characteristic polynomial of the matrix $\phi(\hat{a}_1\lambda_1 + \hat{a}_2\lambda_2 + \cdots + \hat{a}_l\lambda_l)$ by using λ_i as the value for x_i in $\chi(x_1, x_2, \ldots, x_l, t)$, for $1 \leq i \leq l$. For an element $a = \hat{a}_1\lambda_1 + \hat{a}_2\lambda_2 + \cdots \hat{a}_l\lambda_l$ with the property described in Theorem 2.4.20, the minimal polynomial of a has degree l in t, and is squarefree. Since this is also a divisor of the characteristic polynomial $\chi(\lambda_1, \lambda_2, \ldots, \lambda_l, t)$, which also has degree l in t, we conclude that these polynomials are the same, and that $\chi(\lambda_1, \lambda_2, \ldots, \lambda_l, t)$ is squarefree in $F[t]$. Further, we note that for any choice of values $\eta_1, \eta_2, \ldots, \eta_l \in F$, the polynomial $\chi(\eta_1, \eta_2, \ldots, \eta_l, t)$ is squarefree in $F[t]$ only if this is the minimal polynomial of an element $\bar{a} = \hat{a}_1\eta_1 + \hat{a}_2\eta_2 + \cdots + \hat{a}_l\eta_l$ of Centre(A) which can be used (alone) to generate the central primitive idempotents of A by "Extraction of Idempotents".

We recall that, since F is perfect, the polynomial $\chi(\eta_1, \eta_2, \ldots, \eta_l, t)$ is squarefree in $F[t]$ if and only if the greatest common divisor of $\chi(\eta_1, \eta_2, \ldots, \eta_l, t)$ and $\frac{\mathrm{d}}{\mathrm{dt}}\chi(\eta_1, \eta_2, \ldots, \eta_l, t)$ is 1 in $F[t]$. If

$$\psi(x_1, x_2, \ldots, x_l, t) = \frac{\mathrm{d}}{\mathrm{dt}}\chi(x_1, x_2, \ldots, x_l, t),$$

then ψ is a polynomial with total degree at most l in $x_1, x_2, \ldots, x_l$, and with degree less than l in t, such that

$$\gcd(\chi(\lambda_1, \lambda_2, \ldots, \lambda_l, t), \psi(\lambda_1, \lambda_2, \ldots, \lambda_l, t)) = 1,$$

for $\lambda_1, \lambda_2, \ldots, \lambda_l \in F$ as chosen above. We now use the following fact, which is proved by Loos [87] (Theorem 5).

Proposition 2.4.21. Let $A, B \in R[t]$, with $A, B \neq 0$, for an integral domain R and an indeterminate t over R. Then $\mathrm{Res}_t(A, B) = 0$ if and only if $\deg(\gcd(A, B)) > 0$.

Now we write

$$h(x_1, x_2, \ldots, x_l) = \mathrm{Res}_t(\chi(x_1, x_2, \ldots, x_l, t), \psi(x_1, x_2, \ldots, x_l, t)).$$

The polynomial h is a polynomial in $F[x_1, x_2, \ldots, x_l]$ with total degree at most $l(2l-1)$ in the indeterminates $x_1, x_2, \ldots, x_l$. Using Proposition 2.4.21, and the previous remarks about χ and ψ, we obtain the following lemma.

Lemma 2.4.22. Let $\eta_1, \eta_2, \ldots, \eta_l \in F$; then the element $a = \hat{a}_1\eta_1 + \hat{a}_2\eta_2 + \cdots + \hat{a}_l\eta_l$ of Centre(A) can be used as input for the algorithm "Extraction of Idempotents" to generate the central primitive idempotents of A, if and only if $h(\eta_1, \eta_2, \ldots, \eta_l) \neq 0$.

We also use the following fact, which is proved by Schwartz [111] (Corollary 1).

Proposition 2.4.23. Let $I \subseteq F$ and let $|I| \geq ck$. If $f \in F[x_1, x_2, \ldots, x_l]$, with total degree at most k, and f is not identically zero, then the number of zeros of f in $I \times I \times \cdots \times I$ is at most $c^{-1}|I|^l$.

Theorem 2.4.24. Let $\hat{a}_1, \hat{a}_2, \ldots, \hat{a}_l$ be a basis over F for the centre of a finite-dimensional semi-simple associative algebra A over a perfect field F, and let $c > 0$, such that F contains at least $l(2l-1)c$ distinct elements. Let $I \subseteq F$ such that $|I| = l(2l-1)c$. If $\eta_1, \eta_2, \ldots, \eta_l$ are chosen randomly and independently from the set I, then the probability that the element $a = \hat{a}_1\eta_1 + \hat{a}_2\eta_2 + \cdots + \hat{a}_l\eta_l$ cannot be used to generate the central primitive idempotents of A, using the algorithm "Extraction of Idempotents", is at most $1/c$.

Proof. This follows immediately from Theorem 2.4.20, Lemma 2.4.22, and from Proposition 2.4.23. ■

We use this result to obtain the algorithm on the following page.

Theorem 2.4.25. Let F be an infinite perfect field, and let $\epsilon > 0$, and suppose $A \subseteq M_{m\times m}(F)$ is a finite-dimensional semi-simple associative algebra of dimension n over F. Let I be a subset of F of size $\lceil n(2n-1)\epsilon^{-1} \rceil$. Then the algorithm "Simple Components via Primitive Elements" can be used to compute bases for the simple components of A, or to report failure, using arithmetic-Boolean circuits over F (with oracle nodes for factorisation of squarefree polynomials in $F[t]$, and for the selection of random elements of I), of depth $O(\log^2(nm))$ and size $(nm)^{O(1)}$, plus the cost of selecting at most n random elements from I in parallel, and the cost of factoring a single squarefree polynomial, with degree at most n, in $F[t]$. The probability of failure is less than ϵ.

Proof. The correctness of the algorithm, and the upper bound on the probability of failure, are consequences of Theorem 2.4.24. The timing analysis is straightforward. ■

```
Algorithm  Simple Components via Primitive Elements
Input.     • Integers $n$, $m > 0$.
           • Matrices $a_1, a_2, \ldots, a_n \in M_{m \times m}(F)$, which form the basis for a
             finite-dimensional semi-simple associative algebra $A \subseteq M_{m \times m}(F)$
             of dimension $n$ over $F$.
           • Error tolerance $\epsilon > 0$.
Output.    EITHER:
           • Integer $k > 0$, the number of simple components of $A$.
           • Integers $n_1, n_2, \ldots, n_k > 0$, with $n_1 + n_2 + \cdots + n_k = n$,
             such that $n_i$ is the dimension of simple components $B_i$ of $A$.
           • Elements $\mu_{i\,j\,l}$ of $F$, for $1 \leq i \leq k$, $1 \leq j \leq n_i$, and $1 \leq l \leq n$,
             defining elements $b_{i\,j} = \sum_{l=1}^{n} \mu_{i\,j\,l} a_l$ of $A$ such that
               (1)  $b_{i\,1}, b_{i\,2}, \ldots, b_{i\,n_i}$ is a basis for the simple
                      component $B_i$ over $F$, and hence
               (2)  $b_{1\,1}, \ldots, b_{k\,n_k}$ is a basis for $A$ over $F$.
           • Matrices $c_{i\,j} \in M_{n_i \times n_i}(F)$, such that $c_{i\,1}, c_{i\,2}, \ldots, c_{i\,n_i}$
             is the basis for a matrix algebra isomorphic to $B_i$, for $1 \leq i \leq k$.
           OR:  failure,  with probability less than $\epsilon$.

(1)  Compute a basis $\hat{a}_1, \hat{a}_2, \ldots, \hat{a}_l$ for Centre$(A)$ over $F$.
(2)  Choose elements $\lambda_1, \lambda_2, \ldots, \lambda_l$ randomly and independently from
     a subset $I$ of $F$, of size $\lceil l(2l-1)\epsilon^{-1} \rceil$.
(3)  Compute the minimal polynomial in $F[t]$ of the element
       $a = \lambda_1 \hat{a}_1 + \lambda_2 \hat{a}_2 + \cdots + \lambda_l \hat{a}_l$ of Centre$(A)$.
     If this polynomial has degree less than $l$, report failure.
     Otherwise, perform steps 4–5.
(4)  Use the algorithm “Extraction of Idempotents” with input $a$ to compute
     integer $k > 0$ and the primitive idempotents $e_1, e_2, \ldots, e_k$.
(5)  Use the algorithm “Simple Components via Central Primitive Idempotents”
     to compute the remaining values to be generated as output.
```

Thus we can reduce the use of factorisation in computation of simple components over infinite perfect fields to the factorisation of a single squarefree polynomial. (In comparison, each of the algorithms discussed in Sections 2.4.2 and 2.4.3 require the factorisation of $\Omega(n)$ polynomials, each of degree at most m, in order to compute the simple components of a semi-simple algebra $A \subseteq M_{m\times m}(F)$ of dimension n.) This is the best we can do: We cannot decompose semi-simple associative algebras over F using time (significantly) less than that required for factorisation, as indicated by the following theorem.

Theorem 2.4.26. Let F be a field, and suppose the problem "Extraction of Simple Components" can be solved for an arbitrary finite-dimensional semi-simple associative algebra $A \subseteq M_{m\times m}(F)$ of dimension n over F, using arithmetic-Boolean circuits over F of depth $D(m, n)$ and size $S(m, n)$. Then given the coefficients of a squarefree polynomial $f \in F[t]$ of degree n, the irreducible factors of f can be computed using arithmetic-Boolean circuits over F, of depth $D(n, n) + O(\log^2 n)$, and size $S(n, n) + n^{O(1)}$.

Proof. We assume without loss of generality that f is monic; for if F has leading coefficient $c \neq 0$, then the coefficients of the monic polynomial $c^{-1}f$ can be computed using constant depth and size linear in n, while the factors of f can be recovered from those of $c^{-1}f$ at a similar cost. We factor f in $F[t]$ by considering the matrix algebra $A \subseteq M_{n\times n}(F)$, with basis $1, \hat{t}, \hat{t}^2, \dots, \hat{t}^{n-1}$, where $\hat{t} \in M_{n\times n}(F)$ is the companion matrix of f. A is isomorphic to $F[t]/(f)$; since f is squarefree in $F[t]$, A is semi-simple (see Examples 2.1.5 and 2.2.6, and Theorem 2.3.22).

Now suppose $f = g_1 g_2 \cdots g_k$, for irreducible monic polynomials $g_1, g_2, \dots, g_k \in F[t]$. Since $A \cong F[t]/(f)$, and

$$F[t]/(f) \cong F[t]/(g_1) \oplus F[t]/(g_2) \oplus \cdots \oplus F[t]/(g_k),$$

by the Chinese remainder theorem, and since $F[t]/(g_i)$ is an extension field of F, and hence a simple algebra over F, it is clear that A has simple components $B_1, B_2, \dots, B_k$, with $B_i \cong F[t]/(g_i)$ for $1 \leq i \leq k$. Using the bounds given in the statement of the theorem, we see that we can compute bases over F for each of the simple components B_i of A, from the coefficients of f, using arithmetic-Boolean circuits over F of depth $D(n, n) + O(\log^2 n)$ and size $S(n, n) + n^{O(1)}$. The primitive idempotents $e_1, e_2, \dots, e_k$ can be computed from these bases by solving systems of linear equations, using depth $O(\log^2 n)$ and size $n^{O(1)}$. Finally, it is easily checked that the element $\hat{t}e_i$ of B_i has minimal polynomial g_i, for $1 \leq i \leq k$; the coefficients of this polynomial can also be computed (from e_i) at the stated cost, as required. ■

As noted in Section 1.4, there exist fields $\hat{F}$ such that the problem of deciding whether a squarefree polynomial $f \in \hat{F}[t]$ of degree two is irreducible in $\hat{F}[t]$ is an undecidable problem. We conclude from Theorem 2.4.26 that the problem “Extraction of Simple Components” is also undecidable over these fields (even for semi-simple algebras of dimension two over the ground field).

2.4.5. Extraction of Simple Components over $\mathbb{R}$ and $\mathbb{C}$

We now consider the cost of “Extraction of Simple Components” of semi-simple algebras over $\mathbb{R}$ and $\mathbb{C}$. As explained in Section 2.2.3, we assume that we are given an algebra A over a number field F as input; we wish to compute bases for the simple components of $A \otimes_F \mathbb{C}$ over $\mathbb{C}$, or of $A \otimes_F \mathbb{R}$ over $\mathbb{R}$ (if $F \subseteq \mathbb{R}$).

Suppose now that $A \subseteq M_{m\times m}(F)$ is a finite-dimensional semi-simple associative algebra of dimension n over a number field F. We begin by considering the simple components of $A \otimes_F \mathbb{C}$ over $\mathbb{C}$. We will show that it is sufficient to perform computations over a number field $E \supseteq F$ in order to compute bases (over $\mathbb{C}$) for these components.

We first consider the case that A is a simple algebra over F. Now the centre of A is a finite algebraic extension field of F: $\mathrm{Centre}(A) = F[\alpha]$, for some $\alpha \in \mathrm{Centre}(A)$ with minimal polynomial $f \in F[t]$, with f monic and irreducible in $F[t]$. Since f is irreducible, and F is perfect, the roots of f are distinct in $\mathbb{C}$:

$$f = \prod_{i=1}^{h} (t - \alpha_i)$$

for distinct $\alpha_1, \alpha_2, \ldots, \alpha_h$ in $\mathbb{C}$. In fact, these roots all lie in some number field $E \supseteq F$, a *splitting field* for the polynomial f over F.

Since the polynomials $t-\alpha_1$, $t-\alpha_2$, …, $t-\alpha_h$ are pairwise relatively prime in $E[t]$, there exist polynomials $g_1, g_2, \ldots, g_h \in E[t]$, each with degree less than that of f, such that

$$g_i \equiv 1 \pmod{(t-\alpha_i)} \qquad \text{and} \qquad g_i \equiv 0 \pmod{(t-\alpha_j)}$$

for $1 \leq i, j \leq h$, $i \neq j$; or, equivalently,

$$g_i(\alpha_i) = 1 \qquad \text{and} \qquad g_i(\alpha_j) = 0.$$

Fortunately, the coefficients of g_i lie in a smaller number field than E ($[E : F]$ may be as large as $n!$): The above conditions are equivalent to the conditions that each polynomial g_i has degree less than that of f, and that

$$g_i \equiv 1 \pmod{(t-\alpha_i)} \qquad \text{and} \qquad g_i \equiv 0 \pmod{\left(\frac{f}{(t-\alpha_i)}\right)}.$$

Since f and $(t - \alpha_i)$ are both in $(F[\alpha_i])[t]$, and $(t - \alpha_i)$ divides f, it is clear that g_i can also be chosen from $(F[\alpha_i])[t]$. Now it is easily checked that the idempotent $e_i = g_i(\alpha)$ is in $M_{m\times m}(F[\alpha_i])$, for $1 \leq i \leq h$, and that

$$e_1 + e_2 + \cdots + e_h = 1 \qquad \text{and} \qquad e_i e_j = \delta_{i\,j} e_i \text{ for } 1 \leq i,\, j \leq h,$$

in $A \otimes_F \mathbb{C}$. Furthermore, it is easily checked that $(\text{Centre}(A \otimes_F \mathbb{C}))e_i \cong \mathbb{C}$ for $1 \leq i \leq h$, so that in fact e_i is a central primitive idempotent of $A \otimes_F \mathbb{C}$. Now let $B_1, B_2, \ldots, B_h$ be the simple components of $A \otimes_F \mathbb{C}$ with identity elements $e_1, e_2, \ldots, e_h$ respectively. Since $B_i = (A \otimes_F \mathbb{C})e_i$, it is clear that B_i is spanned (over $\mathbb{C}$) by the matrices $a_1 e_i, a_2 e_i, \ldots, a_n e_i$, if $a_1, a_2, \ldots, a_n$ is a basis for A over F. Since these matrices all have entries in $F[\alpha_i]$, it is clear that the simple component B_i of $(A \otimes_F \mathbb{C})$ over $\mathbb{C}$ has a basis and set of structure constants in the number field $F[\alpha_i] \cong F[t]/(f)$, for $1 \leq i \leq h$.

We now consider the more general case, that A is semi-simple over F. Suppose again that A has simple components $B_1, B_2, \ldots, B_k$ over F; then it is clear that $B_1 \otimes_F \mathbb{C}, B_2 \otimes_F \mathbb{C}, \ldots, B_k \otimes_F \mathbb{C}$ are two-sided ideals in $A \otimes_F \mathbb{C}$, and that the set of simple components of $A \otimes_F \mathbb{C}$ includes all of the simple components of $B_i \otimes_F \mathbb{C}$, for $1 \leq i \leq k$. We generate bases for the simple components of $A \otimes_F \mathbb{C}$ by first generating bases over F for the simple components $B_1, B_2, \ldots, B_k$ of A over F, and then computing bases (over $\mathbb{C}$) for the simple components of $B_i \otimes_F \mathbb{C}$, for $1 \leq i \leq k$.

The process of computing the simple components of $A \otimes_F \mathbb{R}$, for a semi-simple algebra $A \subseteq M_{m\times m}(F)$ of dimension n over F, and for a real number field $F \subseteq \mathbb{R}$, is similar to that of computing the simple components of $A \otimes_F \mathbb{C}$. As above, we consider the simple components B_i of A over F separately; we express the centre of B_i as $F[\alpha_i]$, for $\alpha_i \in B_i$, with minimal polynomial $f_i \in F[t]$ monic and irreducible in $F[t]$. Computing the factorisation of f in $\mathbb{R}[t]$, we obtain the expression

$$f_i = f_{i\,1} f_{i\,2} \cdots f_{i\,h_i}$$

for $f_{i\,j} \in \mathbb{R}[t]$ monic and irreducible — hence either of the form $t - \alpha_{i\,j}$, for $\alpha_{i\,j}$ some (real) algebraic number, or of the form $t^2 + \alpha_{i\,j\,1} t + \alpha_{i\,j\,0}$, again for real algebraic numbers $\alpha_{i\,j\,1}, \alpha_{i\,j\,0}$, in some number field $F[\alpha_{i\,j}] = F[\alpha_{i\,j\,1}, \alpha_{i\,j\,0}]$. Again, we compute polynomials $g_{i\,1}, g_{i\,2}, \ldots, g_{i\,h_i}$, with the coefficients of $g_{i\,j}$ in $F[\alpha_{i\,j}] \subseteq \mathbb{R}$, and such that $g_{i\,j} \equiv 1 \pmod{f_{i\,j}}$ and $g_{i\,j} \equiv 0 \pmod{f_{i\,l}}$ for $1 \leq j,\, l \leq h_i$ and for $j \neq l$. The matrix $g_{i\,j}(\alpha_i) = e_{i\,j}$ is a central primitive idempotent in $A \otimes_F \mathbb{R}$, for $1 \leq i \leq k$ and $1 \leq j \leq h_i$. If $f_{i\,j}$ has degree 1 in $\mathbb{R}[t]$ then the centre of the simple component $B_{i\,j} = (A \otimes_F \mathbb{R})e_{i\,j}$ is isomorphic to $\mathbb{R}$; otherwise, $f_{i\,j}$ has degree two, and the centre of the simple component $B_{i\,j} = (A \otimes_F \mathbb{R})e_{i\,j}$ is isomorphic to $\mathbb{C}$. Computation of bases for the simple components of $A \otimes_F \mathbb{R}$ from the central primitive idempotents proceeds as in the case for computations over $\mathbb{C}$.

We will require primitive generators of field extensions of number fields. We use the (well known) fact that if F is a number field, E is an extension of degree at most n over F, and $\alpha, \beta \in E$, then there is some integer c between 0 and n^2 such that $F[\alpha + c\beta] = F[\alpha, \beta] \subseteq E$ (see, for example, van der Waerden [117], Section 6.10). As we explain below, Friedl and Rónyai apply this to obtain a primitive generator of $E = F[a_1, a_2, \ldots, a_n]$ of the form $a_1 + c_2a_2 + \cdots + c_na_n$, where $c_2, c_3, \ldots, c_n$ are small integers, in polynomial time. We apply this in a slightly different way to obtain an efficient parallel algorithm for this computation — but requiring the use of larger multipliers $c_2, c_3, \ldots, c_n$.

Proposition 2.4.27. Let E be an extension field of a number field F, and let $a_1, a_2, \ldots, a_n$ be a basis for E over F.

(i) (Friedl and Rónyai [43]). There exists an element $a = a_1 + c_2a_2 + \cdots + c_na_n \in E$, with $c_i \in \mathbb{Z}$ and $0 \leq c_i \leq n^2$ for $2 \leq i \leq n$, such that $E = F[a]$, and such that the element a can be computed using arithmetic-Boolean circuits over F of size polynomial in n, or using Boolean circuits of size polynomial in the input size N.

(ii) There exists an element $\hat{a} = a_1 + \hat{c}_2a_2 + \cdots + \hat{c}_na_n \in E$, with $\hat{c}_i \in \mathbb{Z}$ and $0 \leq \hat{c}_i \leq n^{2\log_2 n}$ (so that $\hat{c}_i$ has a binary representation of length $O(\log^2 n)$), such that $E = F[\hat{a}]$, and which can be computed using arithmetic-Boolean circuits over F of size $n^{O(1)}$ and depth $O(\log^3 n)$, or using Boolean circuits of size $N^{O(1)}$ and of depth $O(\log^3 N)$, for input size N.

Proof. Given α and β in E, we check whether $\alpha, \beta \in F[\alpha + c\beta]$ for $c \in \mathbb{Z}$ by solving systems of linear equations over F, using arithmetic-Boolean circuits of size $n^{O(1)}$ and depth $O(\log^2 n)$. A generator a with the properties described in (i) can be obtained by computing elements $b_1, b_2, \ldots, b_n \in E$, in sequence, with $b_1 = a_1$, and $b_{i+1} = b_i + c_{i+1}a_{i+1}$, with $0 \leq c_{i+1} \leq n^2$ chosen such that $b_i, a_{i+1} \in F[b_{i+1}]$ (see the procedures SPLIT1 and PRIMELEM and Proposition 7.5 of Friedl and Rónyai [43]). It is easily checked (by induction on n) that the element $a = b_n$ has the desired properties.

We obtain a generator $\hat{a}$ with the properties described in (ii) by using a divide-and-conquer approach: Elements $\hat{a}_1, \hat{a}_2$ of E are generated so that $a_1, a_2, \ldots, a_{\hat{n}} \in F[\hat{a}_1]$ and $a_{\hat{n}+1}, a_{\hat{n}+2}, \ldots, a_n \in F[\hat{a}_2]$, for $\hat{n} = \lceil n/2 \rceil$, and so that

$$\hat{a}_1 = a_1 + \hat{c}_2a_2 + \cdots + \hat{c}_{\hat{n}}a_{\hat{n}},$$

and

$$\hat{a}_2 = a_{\hat{n}+1} + \hat{c}_{\hat{n}+1}a_{\hat{n}+2} + \cdots + \hat{c}_na_n,$$

with coefficients $\hat{c}_i \in \mathbb{Z}$ such that $0 \leq \hat{c}_i \leq n^{2((\log_2 n)-1)}$. An element $\hat{a} = \hat{a}_1 + c\hat{a}_2$, with $0 \leq c \leq n^2$, is then computed so that $\hat{a}_1, \hat{a}_2 \in F[\hat{a}]$. Again, it is easily checked that $\hat{a}$ has the desired properties, and can be computed at the stated cost. ■

We can now state our algorithm for computation of simple components of $A \otimes_F \mathbb{R}$ or of $A \otimes_F \mathbb{C}$. Since the computations are similar for the two cases, we give a single algorithm, "Extraction of Simple Components over an Extension", for computations over an extension E. The algorithm uses factorisation of squarefree polynomials over E, and isolation of roots in E; we assume the use of algorithms in Sections 1.5 and 1.6 for these problems, for the cases $E = \mathbb{R}$ and $E = \mathbb{C}$. The algorithm performs computations over several number fields; for each field F, a single generator α (with $F = \mathbb{Q}[\alpha]$) is either provided as input or is computed. We assume elements of a number field F are represented by polynomials in $\mathbb{Q}[t]$, with the element $h(\alpha)$ represented by the polynomial $h \in \mathbb{Q}[t]$ of smallest possible degree. Thus all the inputs and outputs are represented as (vectors of) elements of $\mathbb{Q}$.

Theorem 2.4.28. Let $A \subseteq M_{m\times m}(F)$ be a finite-dimensional semi-simple associative algebra of dimension n over a number field $F = \mathbb{Q}[\alpha]$, for some algebraic integer α with minimal polynomial $f \in \mathbb{Z}[t]$. Suppose we are given a description of A and F consisting of a basis for A over F, as well as the minimal polynomial f of α, and an isolating region for α in $\mathbb{C}$.

(i) There exists an integer $k \le n$, and number fields $E_1, E_2, \ldots, E_k \subseteq \mathbb{C}$, each an extension of degree at most n over F, and a simple algebra C_i over the number field E_i, for $1 \le i \le k$, such that $A \otimes_F \mathbb{C}$ has simple components $B_1, B_2, \ldots, B_k$ over $\mathbb{C}$, with $B_i \cong C_i \otimes_{E_i} \mathbb{C}$, for $1 \le i \le k$. A description of the number field E_i (including the minimal polynomial over $\mathbb{C}$ and an isolating region in $\mathbb{C}$ for an algebraic integer α_i, with $E_i = \mathbb{Q}[\alpha_i] \supseteq F$), and a basis over E_i for the simple algebra C_i, can be computed, for $1 \le i \le k$, from the description of A and F, using Boolean circuits of polynomial size.

(ii) If $F \subseteq \mathbb{R}$ then there exists an integer $k \le n$ and number fields $E_1, E_2, \ldots, E_k \subseteq \mathbb{R}$, each an extension of degree at most n over F, and a simple algebra C_i over E_i, for $1 \le i \le k$, such that $A \otimes_F \mathbb{R}$ has simple components $B_1, B_2, \ldots, B_k$ over $\mathbb{R}$, with $B_i \equiv C_i \otimes_{E_i} \mathbb{R}$, for $1 \le i \le k$. A description of the number field E_i (as described above) and a basis over E_i for the simple algebra C_i, can be computed for $1 \le i \le k$ from the description of A over F, using Boolean circuits of polynomial size.

Proof. The algorithm "Extraction of Simple Components over an Extension" is used in both (i) and (ii) — for the extension $E = \mathbb{C}$ in the first case, and for $E = \mathbb{R}$ in the second. The methods used in the algorithm are the same as those used in the algorithms "Extraction of Idempotents" and "Simple Components via Central Primitive Idempotents", stated earlier. The correctness of "Extraction of Simple Components over an Extension" follows from the correctness of these methods.

The timing analysis follows from the time bounds given for the algorithms for linear algebra over extensions in Section 1.3, and for factorisation over number fields, and over $\mathbb{R}$ and $\mathbb{C}$, in Sections 1.4–1.6. ■

Algorithm **Extraction of Simple Components over an Extension**

Input.
- Integers l, n, $m > 0$.
- The coefficients of a monic irreducible polynomial $f \in \mathbb{Q}[t]$
  $f = t^l + \lambda_{l-1}t^{l-1} + \cdots + \lambda_1 t + \lambda_0$ of degree l.
- Coordinates of an isolating region in an extension E of $\mathbb{Q}$ for a root $\alpha \in E$ of f.
- Matrices $a_1, a_2, \ldots, a_n \in M_{m\times m}(F)$, for $F = \mathbb{Q}[\alpha]$, which form the basis for a finite-dimensional semi-simple associative algebra $A \subseteq M_{m\times m}(F)$ of dimension n over F.

Output.
- Integer $k > 0$, the number of simple components of $\hat{A} = A \otimes_F E$.
- Integers $n_1, n_2, \ldots, n_k > 0$, with $n_1 + n_2 + \cdots + n_k = n$, such that n_i is the dimension of the simple component B_i of $\hat{A}$.
- For each integer i, such that $1 \le i \le k$:
  - — Integer $l_i > 0$, the coefficients of a monic irreducible polynomial $f_i = t^{l_i} + \lambda_{i,\,l_i-1}t^{l_i-1} + \cdots \lambda_{i,\,1}t + \lambda_{i,\,0} \in \mathbb{Z}[t]$ of degree l_i, and the coordinates of an isolating region in E for a root α_i of f_i, such that $F \subseteq E_i = \mathbb{Q}[\alpha_i] \subseteq E$.
  - — Coefficients of a polynomial
    $g_i = \zeta_{i,\,l_i-1}t^{l_i-1} + \cdots + \zeta_{i,\,1}t + \zeta_{i,\,0} \in \mathbb{Q}[t]$
    such that $\alpha = g_i(\alpha_i)$.
  - — Elements $\mu_{i\,j\,h}$ of E_i, for $1 \le j \le n_i$ and $1 \le h \le n$, defining elements $b_{i\,j} = \sum_{l=1}^{n} \mu_{i\,j\,h} a_h$ of $\hat{A}$ such that
    (i) $b_{i\,1}, b_{i\,2}, \ldots, b_{i\,n_i}$ is a basis for the simple component B_i over E, and hence
    (ii) $b_{1\,1}, \ldots, b_{k\,n_k}$ is a basis for $\hat{A}$ over E.
  - — Matrices $c_{i\,j} \in M_{n_i\times n_i}(E_i)$, such that $c_{i\,1}, c_{i\,2}, \ldots, c_{i\,n_i}$ is a basis for a matrix algebra $C_i \subseteq M_{n_i\times n_i}(E_i)$, such that $C_i \otimes_{E_i} E$ is isomorphic to B_i.

(1) Use the algorithm "Simple Components via Idempotents of Basis" to compute the number $\hat{k}$ of simple components of A over F, as well as positive integers $\hat{n}_1, \hat{n}_2, \ldots, \hat{n}_{\hat{k}}$, with $\hat{n}_1 + \hat{n}_2 + \cdots + \hat{n}_{\hat{k}} = n$, and a basis $\hat{b}_{i\,1}, \hat{b}_{i\,2}, \ldots, \hat{b}_{i\,\hat{n}_i}$ for the simple components $\hat{B}_i$ of A over F, for $1 \le i \le \hat{k}$.

(2) Perform steps 3–5 for each integer i such that $1 \le i \le \hat{k}$.

(3) Compute an element $\hat{\alpha}_i$ of Centre$(\hat{B}_i)$, with minimal polynomial $h_i \in F[t]$, such that Centre$(\hat{B}_i) = F[\hat{\alpha}_i]$.

(4) Use factorisation over E to compute the following values.

- An integer $m_i > 0$, and an element c_i of F, such that there exist monic irreducible polynomials $h_{i\,1}, h_{i\,2}, \ldots, h_{i\,m_i} \in E[t]$, such that $h_i = c_i h_{i\,1} h_{i\,2} \cdots h_{i\,m_i}$;
- The minimal polynomial $\hat{f}_{i\,j} \in \mathbb{Z}[t]$, with degree $\hat{l}_{i\,j}$, and an isolating region in E, for an algebraic integer $\hat{\alpha}_{i\,j} \in E$ such that α and the coefficients of the polynomial $h_{i\,j}$ lie in the number field $\hat{E}_{i\,j} = \mathbb{Q}[\hat{\alpha}_{i\,j}]$ (so $F \subseteq \hat{E}_{i\,j} \subseteq E$), for $1 \leq j \leq m_i$;
- A polynomial $\hat{g}_{i\,j} \in \mathbb{Q}[t]$ with degree less than $\hat{l}_{i\,j}$ such that $\alpha = \hat{g}_{i\,j}(\hat{\alpha}_{i\,j})$ represents α as an element of $\hat{E}_{i\,j}$, and embeds F in $\hat{E}_{i\,j}$;
- Polynomials in $\mathbb{Q}[t]$ representing each of the coefficients of $h_{i\,j}$ as elements of $\hat{E}_{i\,j}$, (as α is represented by the polynomial $\hat{g}_{i\,j}$), for $1 \leq j \leq m_i$.

(5) Compute polynomials $\bar{g}_{i\,j} \in \hat{E}_{i\,j}[t]$, for $1 \leq j \leq m_i$, such that

$$\bar{g}_{i\,j} \equiv 1 \pmod{h_{i\,j}} \qquad \text{and} \qquad \bar{g}_{i\,j} \equiv 0 \pmod{(h_i/h_{i\,j})}.$$

Let $\hat{e}_{i\,j} = \bar{g}_{i\,j}(\hat{\alpha}_i) \in \hat{B}_i \otimes_F E$; $\hat{e}_{i\,j}$ is a primitive idempotent of $A \otimes_F E$.

(6) Let $k = m_1 + m_2 + \cdots + m_{\hat{k}}$. Set $e_1, e_2, \ldots, e_k$ to be the primitive idempotents $\hat{e}_{i\,j}$ (for $1 \leq i \leq \hat{k}$, and $1 \leq j \leq m_i$) computed in step 5, and set f_h, l_h, α_h, and g_h to be the values $\hat{f}_{i\,j}$, $\hat{l}_{i\,j}$, $\hat{\alpha}_{i\,j}$, and $\hat{g}_{i\,j}$, respectively, which correspond to the idempotent $e_h = \hat{e}_{i\,j}$, for $1 \leq h \leq k$. Let $E_h = \mathbb{Q}[\alpha_h]$, so $F \subseteq E_h \subseteq E$, for $1 \leq h \leq k$.

(7) Perform steps 8–10 for each integer i, such that $1 \leq i \leq k$.

(8) Compute the integer n_i and a basis $b_{i\,1}, b_{i\,2}, \ldots, b_{i\,n_i}$ over E_i for $\hat{C}_i = e_i A \subseteq A \otimes_F E_i$, by selecting a maximal linearly independent subset of $e_i a_1, e_i a_2, \ldots, e_i a_n$. (Note that $B_i \cong C_i \otimes_{E_i} E$.)

(9) Compute elements $\mu_{i\,j\,l}$ of E_i, for $1 \leq j \leq n_i$ and $1 \leq l \leq n$, such that $b_{i\,j} = \mu_{i\,j\,1} a_1 + \mu_{i\,j\,2} a_2 + \cdots + \mu_{i\,j\,n} a_n$.

(10) Compute a set of structure constants for the simple algebra $\hat{C}_i$ with respect to the basis $b_{i\,1}, b_{i\,2}, \ldots, b_{i\,n_i}$ over E_i; these are also structure constants for B_i over E, with respect to the same basis. Use these structure constants to compute the matrix

$$c_{i\,j} = \phi_i(b_{i\,j}) \in C_i \subseteq M_{n_i \times n_i}(E_i), \quad \text{for } 1 \leq j \leq n_i,$$

and for ϕ_i the regular matrix representation of $\hat{C}_i$ with respect to the above basis.

2.5. Decompositions of Simple Algebras

We now consider algorithms for the decomposition of finite-dimensional simple algebras over fields. Suppose A is a simple algebra of dimension n over a field F; then, as stated in Theorem 2.1.25, A is isomorphic to $M_{k\times k}(D)$, the ring of $k \times k$ matrices over some division algebra D over F, for some integer $k > 0$; the division algebra D is unique (up to isomorphism). A then has a division algebra isomorphic to D as a subalgebra; henceforth, we consider D to be embedded in A. If the division algebra D has dimension l over F, then it is clear that $n = lk^2$. Furthermore, if $e_{i\,j} \in A$ (for $1 \leq i,\, j \leq k$) are elements of a *standard basis* for $M_{k\times k}(D)$ over D, so that

$$e_{1\,1} + e_{2\,2} + \cdots + e_{k\,k} = 1, \qquad e_{i\,j} \neq 0, \qquad \text{and} \qquad e_{r\,s}e_{t\,u} = \delta_{s\,t}e_{r\,u},$$

for $1 \leq i,\, j,\, r,\, s,\, t,\, u \leq k$ and for $\delta_{r\,s}$ the Kronecker delta (1 if $s = t$, 0 otherwise), and if $d_1, d_2, \ldots, d_l$ is a basis for $D \subseteq A$ over F, then the set of elements $\{\, d_h e_{i\,j} \;:\; 1 \leq h \leq l,\, 1 \leq i,\, j \leq k \,\}$ form a basis for A over F. Further, the elements of these two bases commute: $d_h e_{i\,j} = e_{i\,j} d_h$. (This is made clear by thinking of elements of A as $k \times k$ matrices with entries in D. The matrices "$e_{i\,j}$" have entries in the field $F \subseteq \text{Centre}(D)$, while the elements $d_h \in D$ correspond to diagonal matrices.) We "decompose" the simple algebra A, isolating the division algebra D (and exhibiting the isomorphism $A \cong M_{k\times k}(D) \cong D \otimes_F M_{k\times k}(F)$), [*] by computing bases for $M_{k\times k}(D)$ over D, and D over F, with the above properties. We state the problem "Decomposition of a Simple Algebra" formally on the following page.

As is the case for the problem "Extraction of Simple Components" of Section 2.4, the above problem can be reduced to the computation of a set of idempotents in A. We show that this reduction is useful, both for sequential and for parallel computations, in Section 2.5.1. However, the decomposition of simple algebras differs from the extraction of simple components of semi-simple algebras in other respects. It is not sufficient to consider commutative algebras when decomposing simple algebras, and the problem of deciding whether a simple algebra is a division algebra is not straightforward. We consider this problem for several classes of fields. In Section 2.5.2 we review the positive result of Rónyai [102], [103], that there is an efficient (probabilistic) algorithm for this problem over finite fields. We then consider computations for simple algebras over $\mathbb{C}$ and $\mathbb{R}$, in Sections 2.5.3 and 2.5.4 respectively; we obtain efficient (new) probabilistic algorithms for these cases. To our knowledge, these are the first (probabilistic) algorithms for the decomposition of simple algebras over these fields, which perform exact computations (rather than computing numerical estimates) using a polynomial number of Boolean operations. Finally, we review the negative results of Rónyai ([103], [104]) for decompositions of simple algebras over $\mathbb{Q}$, and discuss the problem of decomposing simple algebras over number fields, in Section 2.5.5.

[*] The tensor product "$\otimes_F$" is discussed in Section 2.2.3.

Problem **Decomposition of a Simple Algebra**

Input.

- Integers n, $m > 0$.
- Matrices $a_1, a_2, \ldots, a_n \in M_{m\times m}(F)$, which form the basis for a finite-dimensional simple associative algebra $A \subseteq M_{m\times m}(F)$ of dimension n over F.

Output.

- Integers k, $l > 0$, such that $A \cong M_{k\times k}(D)$, for a division algebra $D \subseteq A$ of dimension l over F.
- Elements $\mu_{h\,g} \in F$ and $\nu_{i\,j\,g} \in F$, for $1 \leq h \leq l$, $1 \leq i, j \leq k$, and $1 \leq g \leq n$, defining elements $d_h = \sum_{g=1}^{n} \mu_{h\,g} a_g$ and $e_{i\,j} = \sum_{g=1}^{n} \nu_{i\,j\,g} a_g$ of A, such that
  (1) $d_1, d_2, \ldots, d_l$ is a basis for a division algebra $D \subseteq A$ over F;
  (2) the elements $e_{i\,j}$ form a basis for $A \cong M_{k\times k}(D)$ over D, such that
  $$e_{1\,1} + e_{2\,2} + \cdots + e_{k\,k} = 1, \quad \text{and} \quad e_{r\,s}e_{t\,u} = \delta_{s\,t}e_{r\,u}$$
  for $1 \leq r, s, t, u \leq k$;
  (3) $d_h e_{i\,j} = e_{i\,j} d_h$ for $1 \leq h \leq l$ and $1 \leq i, j \leq k$; and,
  (4) the elements $d_h e_{i\,j}$ (for $1 \leq h \leq l$ and $1 \leq i, j \leq k$), which form a basis for A over F.
- Matrices $c_h \in M_{l\times l}(F)$, such that $c_1, c_2, \ldots, c_l$ is a basis for a matrix algebra isomorphic to D over F.

2.5.1. Simple Algebras and Idempotents

We begin by reducing the decomposition of a simple algebra A over F to the computation of a set of primitive idempotents in A. We will show that this reduction is useful for both sequential and parallel computations (see, in particular, Theorem 2.5.3).

We will use the properties of sets of idempotents stated below.

Proposition 2.5.1. Suppose $A \cong M_{k\times k}(D)$ for D a division algebra of dimension l over a field F, and suppose $e_1, e_2, \ldots, e_h$ is a set of idempotents (so that $e_ie_j = \delta_{i\,j}e_i$ for $1 \leq i \leq h$ and $e_1 + e_2 + \cdots e_h = 1$). Then $h \leq k$, and there exist positive integers $m_1, m_2, \ldots, m_h > 0$ such that $m_1 + m_2 + \cdots + m_h = k$, and so that, for $1 \leq i, j \leq h$,

(i)' e_iA is a right ideal of A, with dimension m_ilk over F;
(ii)' Ae_i is a left ideal of A, with dimension m_ilk over F;
(iii)' e_iAe_j is an F-vector space with dimension m_im_jk over F;
(iv)' e_iAe_i is a simple algebra, isomorphic to $M_{m_i\times m_i}(D)$ over F.

If, in addition, $\{e_1, e_2, \ldots, e_h\}$ is a set of primitive idempotents in A, then $h = k$, $m_i = 1$ for $1 \leq i \leq k$, and

(v)' e_iA is an minimal nonzero right ideal of A;
(vi)' Ae_i is an minimal nonzero left ideal of A;
(vii)' $e_iAe_i \cong D$.

Proof. Since $e_1, e_2, \ldots, e_h$ is a set of (nonzero) idempotents of A, it is easily verified that

- e_iA is a right ideal of A;
- Ae_i is a left ideal of A;
- e_iAe_j is an F-vector space; and,
- e_iAe_i is a simple algebra over F.

Further, if e_i is a primitive idempotent in A, then it is clear that e_iA and Ae_i are, respectively, irreducible (that is, minimal nonzero) right and left ideals of A (so they contain no proper right or left ideals, respectively), and that e_iAe_i is a division algebra over F.

We now make use of the isomorphism $A \cong M_{k\times k}(D)$. If e is an idempotent in $M_{k\times k}(D)$, $b \in M_{k\times k}(D)$ such that $eb = e$, and $d \in D$, then $e(bd) = (eb)d = bd$. It is easily checked that any right ideal of A is isomorphic* to $M_{r\times k}(D)$, for $r \leq k$; any left ideal of A is isomorphic to $M_{k\times r}(D)$, for $r \leq k$; and any simple algebra over F of the form eAe for an idempotent e of D is isomorphic to $M_{r\times r}(D)$ (again,

* We refer here to isomorphism of vector spaces over F, as well as isomorphism of modules over D. For a discussion of modules over rings (generalisations of vector spaces over fields), see (for example) Curtis and Reiner [31], or van der Waerden [117].

for $r \le k$). Further, any irreducible right ideal of A is isomorphic to $M_{1\times k}(D)$, any irreducible left ideal of A is isomorphic to $M_{k\times 1}(D)$, and any division algebra of the form eAe, for a primitive idempotent e of A, is isomorphic to D. In the general case that $e_1, e_2, \ldots, e_h$ is a set of (not necessarily primitive) idempotents in A, we set m_i to be the positive integer such that e_i can be expressed as the sum of exactly m_i primitive idempotents; then properties (i)' through (iv)' follow; properties (v)' to (vii)' also hold, if the idempotents $e_1, e_2, \ldots, e_h$ are primitive. ■

There are several properties of the central idempotents of Section 2.4 which the idempotents discussed here lack. With the exception of the idempotents 0 and 1, the idempotents of a simple algebra A do not belong to the centre of A. They are not unique. Furthermore, a set of primitive idempotents $e_{1\,1}, e_{2\,2}, \ldots, e_{k\,k}$ does not uniquely determine the remaining elements of a standard basis $\{\, e_{i\,j} \,:\, 1 \le i,\, j \le k \,\}$ for $A \cong M_{k\times k}(D)$ over D.

Example 2.5.2. Consider the simple algebra $A = M_{n\times n}(F)$ over F. Clearly we can use as our "standard basis" the canonical basis $e_{i\,j}$, for $1 \le i,\, j \le n$, with $e_{i\,j}$ the matrix in $M_{n\times n}(F)$ whose $(i,j)^{\text{th}}$ entry is 1, and whose remaining entries are 0. Then $e_{1\,1} + e_{2\,2} + \cdots e_{n\,n} = 1$, $e_{i\,i}$ is a primitive idempotent, and $e_{r\,s}e_{t\,u} = \delta_{s\,t}e_{r\,u}$ for $1 \le i,\, r,\, s,\, t,\, u \le n$, as required. However, if X is any nonsingular matrix then it is clear that the matrices $\hat{e}_{i\,j} = X^{-1}e_{i\,j}X$, for $1 \le i,\, j \le n$, also comprise a standard basis. Hence this basis is not unique. In particular, suppose X is a nonsingular diagonal matrix with nonzero diagonal entries $d_1, d_2, \ldots, d_n \in F$. Then $\hat{e}_{i\,i} = e_{i\,i}$ for $1 \le i \le n$, but $\hat{e}_{i\,j} = X^{-1}e_{i\,j}X = d_i^{-1}d_je_{i\,j}$; so $\hat{e}_{i\,j}$ is not generally equal to $e_{i\,j}$ in $M_{n\times n}(F)$, even when $e_{i\,i} = \hat{e}_{i\,i}$ for all i.

In general, if $i \ne j$ then the element "$e_{i\,j}$" of a standard basis for $A \cong M_{k\times k}(D)$ is an element of the algebra $(e_{i\,i} + e_{j\,j})A(e_{i\,i} + e_{j\,j})$ (of dimension 4 over D, and isomorphic to $M_{2\times 2}(D)$) such that $e_{i\,i}e_{i\,j} = e_{i\,j}e_{j\,j} = e_{i\,j}$, and $e_{i\,j}e_{i\,i} = e_{j\,j}e_{i\,j} = 0$. It is evident that $e_{i\,j}$ is determined only to within a multiplicative factor (by an element of D). On the other hand, if the corresponding element $e_{j\,i}$ has been determined, then the element $e_{i\,j}$ is completely determined by the above identities, as well as the conditions $e_{i\,j}e_{j\,i} = e_{i\,i}$, and $e_{j\,i}e_{i\,j} = e_{j\,j}$. It is also evident that, given the primitive idempotents $e_{1\,1}, e_{2\,2}, \ldots, e_{k\,k}$, the elements $e_{1\,2}, e_{1\,3}, \ldots, e_{1\,k}$ can be chosen independently. The elements $e_{2\,1}, e_{3\,1}, \ldots, e_{k\,1}$ will then be fixed. The remaining elements $e_{i\,j}$ will then be uniquely determined, as well, by the relation $e_{i\,j} = e_{i\,1}e_{1\,j}$. It is then easily verified that $e_{r\,s}e_{t\,u} = \delta_{s\,t}e_{r\,u}$, as required.

We must also generate a basis for D over F. We have noted that $e_{1\,1}Ae_{1\,1} \cong D$; we can obtain a basis $\hat{d}_1, \hat{d}_2, \ldots, \hat{d}_l$ of $e_{1\,1}Ae_{1\,1}$ over F by solving systems of linear equations over F. Now we must find elements $d_1, d_2, \ldots, d_l \in D \subseteq A$ which form a basis for the division algebra D over F, so that each $d \in D$ commutes with each element $e_{i\,j}$, and which have the further properties that $e_{1\,1}d_ie_{1\,1} = \hat{d}_i$ for $1 \le i \le l$, and that the elements $e_{j\,j}d_1e_{j\,j}$, $e_{j\,j}d_2e_{j\,j}$, …, $e_{j\,j}d_le_{j\,j}$ form a basis for the division algebra $e_{j\,j}Ae_{j\,j} \cong D$, for $1 \le j \le k$ (so that, in particular, there is an isomorphism of division algebras over F, $\phi_i : D \to e_{i\,i}Ae_{i\,i}$, such that $\phi_i(d_j) = e_{i\,i}d_je_{i\,i}$ for $1 \le j \le l$, and for $1 \le i \le k$). In fact, the element d_i is easily computed from the element $\hat{d}_i$ of $e_{1\,1}Ae_{1\,1}$, for $1 \le i \le l$. We simply note that d_i commutes with $e_{r\,s}$ for $1 \le r$, $s \le k$, and so

$$\begin{aligned} d_i &= (e_{1\,1} + e_{2\,2} + \cdots e_{k\,k})d_i(e_{1\,1} + e_{2\,2} + \cdots e_{k\,k}) \\ &= e_{1\,1}d_ie_{1\,1} + e_{2\,2}d_ie_{2\,2} + \cdots + e_{k\,k}d_ie_{k\,k} \\ &= e_{1\,1}d_ie_{1\,1} + e_{2\,1}(e_{1\,1}d_ie_{1\,1})e_{1\,2} + e_{3\,1}(e_{1\,1}d_ie_{1\,1})e_{1\,3} + \cdots + e_{k\,1}(e_{1\,1}d_ie_{1\,1})e_{1\,k} \\ &= \hat{d}_i + e_{2\,1}\hat{d}_ie_{1\,2} + e_{3\,1}\hat{d}_ie_{1\,3} + \cdots + e_{k\,1}\hat{d}_ie_{1\,k}, \end{aligned}$$

and so d_i is the sum of k terms, each the product of $\hat{d}_i$ and at most two of the elements $e_{r\,s}$, for $1 \le r$, $s \le k$.

We use these facts to decompose a simple algebra A using a set of primitive idempotents, as shown in the algorithm given below.

Algorithm **Decomposition from Primitive Idempotents**

Input.
- Integers n, m, $k > 0$.
- Matrices $a_1, a_2, \ldots, a_n \in M_{m\times m}(F)$, which form the basis for a finite-dimensional simple associative algebra $A \subseteq M_{m\times m}(F)$ of dimension n over F.
- Matrices $e_1, e_2, \ldots, e_k \in A$, which are primitive idempotents in A, such that $e_ie_j = \delta_{i\,j}e_i$ and $e_1 + e_2 + \cdots + e_k = 1$.

Output.
- Integer $l > 0$, such that $A \equiv M_{k\times k}(D)$, for a division algebra $D \subseteq A$ of dimension l over F.
- Elements $\mu_{h\,g} \in F$ and $\nu_{i\,j\,g} \in F$, for $1 \le h \le l$, $1 \le i$, $j \le k$, and $1 \le g \le n$, defining elements $d_h = \sum_{g=1}^{n} \mu_{h\,g}a_g$ and $e_{i\,j} = \sum_{g=1}^{n} \nu_{i\,j\,g}a_g$ of A, such that
  (1) $d_1, d_2, \ldots, d_l$ is a basis for a division algebra $D \subseteq A$ over F;
  (2) the elements $e_{i\,j}$ form a basis for $A \cong M_{k\times k}(D)$ over D, such that
  $e_{1\,1} + e_{2\,2} + \cdots + e_{k\,k} = 1$, and $e_{r\,s}e_{t\,u} = \delta_{s\,t}e_{r\,u}$
  for $1 \le r$, s, t, $u \le k$;

(3) $d_h e_{i\,j} = e_{i\,j} d_h$ for $1 \leq h \leq l$ and $1 \leq i,\, j \leq k$; and,

(4) the elements $d_h e_{i\,j}$ (for $1 \leq h \leq l$ and $1 \leq i,\, j \leq k$) form a basis for A over F.

- Matrices $c_h \in M_{l \times l}(F)$, such that $c_1, c_2, \ldots, c_l$ is a basis for a matrix algebra isomorphic to D over F.

Computation of a Standard Basis

(1) Set $e_{i\,i} = e_i$, for $1 \leq i \leq k$. Perform steps 2–4 for all i such that $2 \leq i \leq k$.

(2) Compute a basis over F for the subspace $(e_{1\,1} + e_{i\,i})A(e_{1\,1} + e_{i\,i})$ of A. Note that this is a simple algebra isomorphic to $M_{2\times 2}(D)$ over F, with identity element $e_{1\,1} + e_{i\,i}$.

(3) Use the above basis to form and solve a system of linear equations over F, computing as $e_{1\,i}$ any nonzero element x of $(e_{1\,1} + e_{i\,i})A(e_{1\,1} + e_{i\,i})$ such that

$$e_{1\,1}x = x, \quad e_{i\,i}x = 0, \quad xe_{1\,1} = 0, \quad \text{and} \quad xe_{i\,i} = x.$$

(4) Form and solve a system of linear equations over F, computing as $e_{i\,1}$ the unique element y of $(e_{1\,1} + e_{i\,i})A(e_{1\,1} + e_{i\,i})$ satisfying the equations

$$e_{1\,1}y = 0, \quad e_{i\,i}y = y, \quad ye_{1\,1} = y, \quad ye_{i\,i} = 0,$$
$$ye_{1\,i} = e_{i\,i}, \quad \text{and} \quad e_{1\,i}y = e_{1\,1}.$$

(5) For $2 \leq i,\, j \leq k$, and $i \neq j$, set $e_{i\,j} = e_{i\,1}e_{1\,j}$.

(6) For $1 \leq i,\, j \leq k$ and $1 \leq g \leq n$, form and solve a nonsingular system of linear equations, to compute the elements $\nu_{i\,j\,g}$ of F such that

$$e_{i\,j} = \textstyle\sum_{g=1}^{n} \nu_{i\,j\,g} a_g.$$

Computation of a Basis for D

(7) Set $l = n/k^2$.

(8) Compute a basis $\hat{d}_1, \hat{d}_2, \ldots, \hat{d}_l$ over F for the subspace $e_{1\,1}Ae_{1\,1}$ of A. Note that $e_{1\,1}Ae_{1\,1}$ is a division algebra isomorphic to D over F.

(9) For all h such that $1 \leq h \leq l$, set $d_h = \hat{d}_h + \sum_{i=2}^{k} e_{i\,1}\hat{d}_h e_{1\,i}$.

(10) For $1 \leq h \leq l$, compute the elements $\mu_{h\,g}$ of F such that $d_h = \sum_{g=1}^{n} \mu_{h\,g} a_g$, by forming and solving a nonsingular system of linear equations over F.

(11) Solving systems of linear equations over F, compute a set of structure constants for the division algebra D, with respect to the basis $d_1, d_2, \ldots, d_l$ over F. Use these constants to compute matrices $c_1, c_2, \ldots, c_l \in M_{l\times l}(F)$, with $c_i = \phi(d_i)$, for $1 \leq i \leq l$, for ϕ the regular matrix representation of D with respect to this basis.

Theorem 2.5.3. Suppose $A \cong M_{k\times k}(D)$, for $k > 0$ and for D a division algebra of dimension $l > 0$ over a field F. Then given a basis for the simple algebra A over F, and given a set of primitive idempotents $e_1, e_2, \ldots, e_k$ of A, the output of the problem "Decomposition of a Simple Algebra" (which would be obtained using the basis for A as input) can be computed using arithmetic-Boolean circuits over F of size $(mn)^{O(1)}$ and depth $O(\log^2(mn))$.

Proof. We use the procedure on the preceding pages to compute the desired output. We first show that this procedure is correct. The first six steps of the procedure compute elements $e_{i\,i} = e_i$, which are given as primitive idempotents, and elements $e_{1\,i}$ and $e_{i\,1}$ such that $e_{1\,1}e_{1\,i} = e_{1\,i}e_{i\,i} = e_{1\,i}$, $e_{i\,1}e_{1\,1} = e_{i\,i}e_{i\,1} = e_{i\,1}$, $e_{i\,i}e_{1\,i} = e_{1\,i}e_{1\,1} = e_{1\,1}e_{i\,1} = e_{i\,1}e_{i\,i} = 0$, $e_{i\,1}e_{1\,i} = e_{i\,i}$, and $e_{1\,i}e_{i\,1} = e_{1\,1}$, for $2 \le i \le n$. Now suppose $2 \le j \le n$ and $i \ne j$; then $e_{j\,j}e_{1\,i} = e_{j\,j}(e_{1\,1}e_{1\,i}) = (e_{j\,j}e_{1\,1})e_{1\,i} = 0$. It follows by a similar argument that $e_{1\,i}e_{j\,j} = e_{j\,j}e_{i\,1} = e_{i\,1}e_{j\,j} = 0$.

Now for $2 \le i, j \le n$, with $i \ne j$, the element $e_{i\,j}$ is set to be $e_{i\,1}e_{1\,j}$; checking $e_{r\,s}e_{t\,u}$, we see that

$$\begin{aligned} e_{r\,s}e_{t\,u} &= e_{r\,1}e_{1\,s}e_{t\,1}e_{1\,u} \\ &= e_{r\,1}(e_{1\,s}e_{s\,s})e_{t\,1}e_{1\,u} \\ &= e_{r\,1}e_{1\,s}(e_{s\,s}e_{t\,1})e_{1\,u} \\ &= \delta_{s\,t}e_{r\,1}e_{1\,s}e_{s\,1}e_{1\,u} \\ &= \delta_{s\,t}e_{r\,u}, \qquad\qquad \text{as desired.} \end{aligned}$$

Thus the elements $e_{i\,j}$ computed in the first six steps have the desired properties.

We now consider the rest of the procedure. We must show that the elements $d_1, d_2, \ldots, d_l$ computed here satisfy the following properties.

- $d_1, d_2, \ldots, d_l$ is a basis over F for a division algebra (isomorphic to D);
- $d_h e_{i\,j} = e_{i\,j}d_h$ for $1 \le h \le l$ and $1 \le i, j \le k$;
- the elements $d_h e_{i\,j}$ (for $1 \le h \le l$ and $1 \le i, j \le k$) form a basis over F for A.

Consider the elements $\hat{d}_1, \hat{d}_2, \ldots, \hat{d}_l$ of $e_{1\,1}Ae_{1\,1}$ computed in step 8 of the procedure. It is a consequence of Proposition 2.5.1 that these form a basis for a division algebra isomorphic to D over F. We show that $d_1, d_2, \ldots, d_l$ also form a basis for a division algebra isomorphic to D, by showing that these elements are linearly independent over F, and by exhibiting an isomorphism of algebras over F, from $e_{1\,1}Ae_{1\,1}$ to the F-vector space with basis $d_1, d_2, \ldots, d_l$, which takes $\hat{d}_i$ to d_i, for $1 \le i \le l$.

Suppose first that $d_1, d_2, \ldots, d_l$ are linearly dependent over F; then there exist elements $\gamma_1, \gamma_2, \ldots, \gamma_l$ of F, not all zero, such that

$$\gamma_1 d_1 + \gamma_2 d_2 + \cdots + \gamma_l d_l = 0.$$

It follows, then, that

$$e_{1\,1}(\gamma_1 d_1 + \gamma_2 d_2 + \cdots + \gamma_l d_l)e_{1\,1} = 0,$$

and so

$$\gamma_1(e_{1\,1}d_1e_{1\,1}) + \gamma_2(e_{1\,1}d_2e_{1\,1}) + \cdots + \gamma_l(e_{1\,1}d_le_{1\,1}) = 0.$$

Now

$$\begin{aligned} e_{1\,1}d_ie_{1\,1} &= e_{1\,1}(\hat{d}_i + e_{2\,1}\hat{d}_ie_{1\,2} + e_{3\,1}\hat{d}_ie_{1\,3} + \cdots + e_{k\,1}\hat{d}_ie_{1\,k})e_{1\,1} \\ &= e_{1\,1}\hat{d}_ie_{1\,1} = e_{1\,1}e_{2\,1}\hat{d}_ie_{1\,2}e_{1\,1} + \cdots e_{1\,1}e_{k\,1}\hat{d}_ie_{1\,k}e_{1\,1} \\ &= e_{1\,1}\hat{d}_ie_{1\,1} = \hat{d}_i, \end{aligned}$$

for $1 \leq i \leq k$, since $\hat{d}_i \in e_{1\,1}Ae_{1\,1}$ and $e_{1\,1}^2 = e_{1\,1}$. It follows that

$$\gamma_1\hat{d}_1 + \gamma_2\hat{d}_2 + \cdots + \gamma_l\hat{d}_l = 0,$$

contradicting the fact that $\hat{d}_1, \hat{d}_2, \ldots, \hat{d}_l$ are linearly independent (over F) in $e_{1\,1}Ae_{1\,1}$. Now suppose

$$\hat{d}_i\hat{d}_j = \sum_{h=1}^{l} \gamma_{i\,j\,h}\hat{d}_h, \qquad \text{for } \gamma_{i\,j\,h} \in F, \quad 1 \leq i, j \leq n,$$

so that the elements $\gamma_{i\,j\,h}$ form a set of structure constants for $e_{1\,1}Ae_{1\,1}$ with respect to the basis $\hat{d}_1, \hat{d}_2, \ldots, \hat{d}_l$. Then

$$\begin{aligned} d_id_j &= \left(\sum_{r=1}^{k} e_{r\,1}\hat{d}_ie_{1\,r}\right)\left(\sum_{s=1}^{k} e_{s\,1}\hat{d}_je_{1\,s}\right) \\ &= \sum_{r=1}^{k}\sum_{s=1}^{k}\Big(e_{r\,1}\hat{d}_i(e_{1\,r}e_{s\,1})\hat{d}_je_{1\,s}\Big) \\ &= \sum_{r=1}^{k} e_{r\,1}\hat{d}_ie_{1\,1}\hat{d}_je_{1\,r} \\ &= \sum_{r=1}^{k} e_{r\,1}\hat{d}_i\hat{d}_je_{1\,r} \qquad\qquad (\text{since } \hat{d}_j \in e_{1\,1}Ae_{1\,1}) \\ &= \sum_{r=1}^{k}\sum_{h=1}^{l} e_{r\,1}\gamma_{i\,j\,h}\hat{d}_he_{1\,r} \\ &= \sum_{h=1}^{l}\gamma_{i\,j\,h}\left(\sum_{r=1}^{k} e_{r\,1}\hat{d}_he_{1\,r}\right) \\ &= \sum_{h=1}^{l}\gamma_{i\,j\,h}d_h, \end{aligned}$$

so that d_1, d_2, $\ldots$, d_h form the basis for an algebra over F, with the same set of structure constants as $e_{1\,1}Ae_{1\,1}$ with respect to $\hat{d}_1$, $\hat{d}_2$, $\ldots$, $\hat{d}_l$. It follows, then, that there is an algebra isomorphism from $e_{1\,1}Ae_{1\,1}$ to this algebra, taking $\hat{d}_i$ to d_i, for $1 \leq i \leq l$, and that this algebra is isomorphic to D.

Now let h, i, $j \in \mathbb{Z}$ such that $1 \leq h \leq l$ and $1 \leq i$, $j \leq k$; clearly

$$\begin{aligned} d_h e_{i\,j} &= \left(\sum_{r=1}^{k} e_{r\,1}\hat{d}_h e_{1\,r} \right) e_{i\,j} \\ &= \sum_{r=1}^{k} e_{r\,1}\hat{d}_h (e_{1\,r}e_{i\,j}) \\ &= e_{i\,1}\hat{d}_h e_{1\,j} \\ &= (e_{i\,j}e_{j\,1})\hat{d}_h e_{1\,j} \\ &= \sum_{s=1}^{k} (e_{i\,j}e_{s\,1})\hat{d}_h e_{1\,s} \\ &= e_{i\,j}\sum_{s=1}^{k} (e_{s\,1}\hat{d}_h e_{1\,s}) \\ &= e_{i\,j}d_h; \end{aligned}$$

so $e_{i\,j}$ and d_h commute.

It follows that $d_h e_{i\,j} = d_h e_{i\,1}e_{1\,j} = e_{i\,1}d_h e_{1\,j} \in e_{i\,1}Ae_{1\,j}$. Further, we can use an argument similar to the one used above to show that d_1, d_2, $\ldots$, d_l are linearly independent, to show that the elements $e_{i\,1}d_1e_{1\,j}$, $e_{i\,1}d_2e_{1\,j}$, $\ldots$, $e_{i\,1}d_le_{1\,j}$ (that is, the elements $d_1e_{i\,j}$, $d_2e_{i\,j}$, $\ldots$, $d_le_{i\,j}$) are linearly independent over F. Now if $\alpha_{i\,j\,h} \in F$ for $1 \leq i$, $j \leq k$ and $1 \leq h \leq l$, and

$$\sum_{i=1}^{k}\sum_{j=1}^{k}\sum_{h=1}^{l} \alpha_{i\,j\,h} d_h e_{i\,j} = 0,$$

then for $1 \leq r$, $s \leq k$, we have

$$e_{r\,r}\left(\sum_{i=1}^{k}\sum_{j=1}^{k}\sum_{h=1}^{l} \alpha_{i\,j\,h} d_h e_{i\,j} \right) e_{s\,s} = 0,$$

so

$$\sum_{h=1}^{l} \alpha_{r\,s\,h} d_h e_{r\,s} = 0,$$

and hence $\alpha_{r\,s\,h} = 0$ for $1 \leq h \leq l$. We conclude that the elements $d_h e_{i\,j}$ are linearly independent over F, and form a basis for A, as required.

Since each step of the procedure requires either the solution of a small system of linear equations over F, or matrix multiplication over F, it is clear that the algorithms can both be implemented using arithmetic-Boolean circuits over F of the size and depth stated in the theorem, as required. ■

We have now reduced the problem of decomposing simple algebras over fields to the computation of a set of primitive idempotents in these algebras. In the next sections we consider the computations of these idempotents in simple algebras over "computable" fields — finite fields, $\mathbb{C}$, $\mathbb{R}$, and $\mathbb{Q}$, as well as the decision problem of whether nontrivial idempotents exist in these algebras.

2.5.2. Simple Algebras over Finite Fields

In this section we review the algorithm of Rónyai ([102], [103]) for the decomposition of simple algebras over finite fields. Rónyai shows that this problem can be solved using a probabilistic algorithm, in polynomial time.

The decision problem, deciding whether a simple algebra over a field $F = \mathbb{F}_{p^c} = \mathbb{F}_q$ is a division algebra, has a straightforward solution.

Theorem 2.5.4. (Wedderburn). A finite division ring is a commutative field.

Corollary 2.5.5. A finite-dimensional simple algebra A over a finite field $\mathbb{F}_q$ is a division algebra if and only if A is commutative.

A proof of Theorem 2.5.4 can be found (for example) in Herstein's monograph on noncommutative rings ([62]). Since any finite-dimensional division algebra over a finite field is a finite division ring, Theorem 2.5.4 implies that any finite-dimensional division algebra over $\mathbb{F}_q$ is commutative. Conversely, since any finite-dimensional simple algebra over $\mathbb{F}_q$ which is not a division algebra is isomorphic to $M_{k\times k}(D)$, for a division algebra D over $\mathbb{F}_q$ and for $k > 1$, and includes noncentral idempotents, it is clear that any finite-dimensional simple algebra over $\mathbb{F}_q$ is commutative if and only if it is a division algebra.

In order to decompose a simple algebra A over $\mathbb{F}_q$, we must also exhibit an isomorphism between A and $M_{k\times k}(E)$, for $E = \mathbb{F}_r$ a finite algebraic extension of $\mathbb{F}_q$, as described at the beginning of Section 2.5. Rónyai adapts the "almost constructive" proof of Wedderburn's Theorem given by Herstein, to obtain a probabilistic Boolean algorithm which performs this computation in polynomial time. We now sketch this algorithm.

Henceforth we consider algebras, and perform computations, over the field $E = \text{Centre}(A) \cong \mathbb{F}_r$. (Note that A is a simple algebra over E, as well as over F.) We can compute an element $\alpha \in \text{Centre}(A)$ such that $E = F[\alpha]$ efficiently using a probabilistic algorithm (by choosing α randomly in E, then verifying that the minimal polynomial of α over E has sufficiently large degree). We can then implement arithmetic over $E = F[\alpha]$ using operations in F, as discussed in Section 1.

As we noted in Section 2.5.1, it is sufficient to generate a set of primitive idempotents $e_1, e_2, \ldots, e_k$ in A in order to decompose the simple algebra A. In fact, it suffices to find an efficient algorithm for the computation of any idempotent $e \notin \{0, 1\}$ in A: for e and $1 - e$ are respectively the identity elements of simple algebras eAe and $(1 - e)A(1 - e)$ over $\mathbb{F}_q$. If $\bar{e}_1, \bar{e}_2, \ldots, \bar{e}_{k_1}$ are primitive idempotents in eAe, so that $\bar{e}_i\bar{e}_j = \delta_{ij}\bar{e}_i$ and $\bar{e}_1 + \bar{e}_2 + \cdots + \bar{e}_{k_1} = e$, and $\hat{e}_1, \hat{e}_2, \ldots, \hat{e}_{k_2}$ are primitive idempotents in $(1-e)A(1-e)$, so that $\hat{e}_i\hat{e}_j = \delta_{ij}\hat{e}_i$ and $\hat{e}_1 + \hat{e}_2 + \cdots + \hat{e}_{k_2} = 1 - e$, then $k = k_1 + k_2$ and we can use $\bar{e}_1, \bar{e}_2, \ldots, \bar{e}_{k_1}, \hat{e}_1, \hat{e}_2, \ldots, \hat{e}_{k_2}$ as a set of primitive idempotents in A, with the algorithms of Section 2.5.1, to decompose A. (Note that $\bar{e}_i\hat{e}_j \in (eAe)((1-e)A(1-e)) = eA(e(1-e))A(1-e) = (0)$, so $\bar{e}_i\hat{e}_j = 0$. Similarly, $\hat{e}_i\bar{e}_j = 0$.)

Thus, it is sufficient to compute an idempotent other than 0 or 1 in A, or to determine that no such idempotents exist (by verifying that A is simple and commutative). We reduce the problem further by showing that we can compute a suitable idempotent from any element a of A such that the minimal polynomial of a over $E \cong \mathbb{F}_r$ is reducible in $E[t]$. In particular, any nonzero zero divisor $a \in A$ has a reducible minimal polynomial over E, and can be used.

Suppose now that $a \in A$ has minimal polynomial f over E, which is reducible in $E[t]$. Suppose first that f is not squarefree, and let g be the squarefree part of f. Then $g(a)$ is nonzero and nilpotent, and a nonzero zero divisor in A. Now $g(a)A$ is a proper right ideal of A. Since $A \cong M_{k\times k}(E)$, there exists some element e of $g(a)A$ such that $ex = x$ for all $x \in g(a)A$. In particular, $e^2 = e$. This idempotent can be computed by solving a system of linear equations over F.

Suppose now that f is squarefree; then $E[a] \cong E[t]/(f)$ is a commutative, semisimple subalgebra of A, and the algorithm "Extraction of Idempotents" of Section 2.4 can be used to compute a set of central primitive idempotents $\tilde{e}_1, \tilde{e}_2, \ldots, \tilde{e}_h$ in $E[a]$, with $\tilde{e}_i \notin \{0, 1\}$ and $h > 1$ (since f is reducible in $E[t]$). While these idempotents are generally neither central nor primitive in A, they are suitable for our purposes.

Wedderburn's Theorem is proved (in Herstein) by a proof of the existence of a nonzero zero divisor in an arbitrary noncommutative finite-dimensional simple algebra over $\mathbb{F}_q$. Rónyai adapts this proof, to obtain a procedure "CUT", which takes such an algebra A as input, and either produces a nontrivial zero divisor in A, or returns a basis over $\mathbb{F}_q$ for a noncommutative proper subalgebra $\hat{A}$ of A, which is also a noncommutative algebra over E. If a zero divisor is produced, then an idempotent can be obtained as indicated above. If a subalgebra $\hat{A}$ is returned, then this subalgebra is examined. Using the methods of Section 2.3, a basis for the radical of $\hat{A}$ is computed. If this is nonzero, then an element y of $\text{rad}(\hat{A})$, nilpotent in A, is obtained; a suitable idempotent is obtained from the proper right ideal yA of A. Otherwise, $\hat{A}$ is semi-simple, and the algorithms of Section 2.4 can be applied; either a suitable idempotent is obtained, or it is determined that $\hat{A}$ is simple. In the latter case, the procedure CUT can be applied recursively to the algebra $\hat{A}$. Now we note that if A and $\hat{A}$ are both simple noncommutative algebras over $\text{Centre}(A)$, and $\hat{A}$ is a proper subalgebra of A, then $A \cong M_{k\times k}(\mathbb{F}_r)$ and $\hat{A} \cong M_{h\times h}(\mathbb{F}_r)$, for $n \ge k > h \ge 2$. It is clear that procedure CUT cannot return proper subalgebras $A_0 = A \supset A_1 = \hat{A} \supset A_2 \supset \cdots$ indefinitely. In particular, this procedure will generated a nontrivial zero divisor of A after at most k iterations.

Let $F = \mathbb{F}_q$, A, and $E = \text{Centre}(A) \cong \mathbb{F}_r$ be as above. Rónyai adapts a lemma stated by Herstein, to prove the following facts.

Proposition 2.5.6. Let $a \in A \setminus \text{Centre}(A)$, such that the minimal polynomial of a over E is irreducible, with degree $s > 0$. Then there exists $c \in A$ such that

(i) $c^{-1}ac = a^r$;

(ii) if $\text{Alg}(a, c)$ denotes the E-subalgebra of A generated by a and c then $\text{Alg}(a, c)$ is a noncommutative algebra over E (and, hence, a noncommutative algebra over F);

(iii) $\text{Alg}(a, c) = E[a] + cE[a] + c^2E[a] + \cdots + c^nE[a] + \cdots$, where $+$ stands for the (not necessarily direct) sum of E-subspaces.

Proposition 2.5.6 is proved by Rónyai, (Lemma 2.1 in [102], Lemma 5.1 in [103]).

Suppose now that an element a has been selected from $A\setminus\text{Centre}(A)$. If the minimal polynomial of a over E is reducible in $E[t]$, then "Extraction of Idempotents" can be used to compute an idempotent in $E[a]$. Otherwise, Proposition 2.5.6 can be applied, and we can solve a system of linear equations to obtain a nonzero element c of A such that $ac = ca^r$. If c is not a unit then c is a zero divisor in A; if the minimal polynomial of c over E is reducible in $E[t]$, then, again, c can be used to produce a nontrivial idempotent in A. Otherwise, c is a unit in A with an irreducible minimal polynomial over E, such that property (i) of Proposition 2.5.6 holds. Since properties (ii) and (iii) are both implied by this first property, we have computed elements a and c with (all of) the properties stated in Proposition 2.5.6, if we have not already found a suitable idempotent in A.

Now a basis for $\text{Alg}(a, c)$ over $\mathbb{F}_q$ is easily computed. Either $\text{Alg}(a, c) \ne A$, and $\text{Alg}(a, c)$ can be returned by the procedure CUT as a noncommutative simple proper subalgebra of A over $\text{Centre}(A)$, or $\text{Alg}(a, c) = A$, and we can use the following facts.

Proposition 2.5.7. (Rónyai). If $\mathrm{Alg}(a,c) = A \cong M_{k\times k}(E)$ then the minimal polynomial of a over E has degree $s = k$ in $E[t]$, $E[c]$ has dimension k over E, $c^k \in E$, and

$$A = \mathrm{Alg}(a,c) = E[a] \oplus cE[a] \oplus c^2E[a] \oplus \cdots \oplus c^{k-1}E[a].$$

Proposition 2.5.7 is proved by Rónyai (see Lemma 2.2 in [102], and Lemma 5.2 in [103]). It follows from Proposition 2.5.7 that if $\mathrm{Alg}(a,c) = A$, then $c^k = \lambda$ for some $\lambda \in E$. Now if the procedure CUT has not already generated a zero divisor or subalgebra of A, then c has an irreducible minimal polynomial over E. By Proposition 2.5.7, this polynomial has degree k, so that $E[c] \cong E[a] \cong \mathbb{F}_{r^k}$, and c has minimal polynomial $t^k - \lambda$ in $E[t]$.

At this point, we recall the notion of *norms* of elements of field extensions (defined formally in Definition 1.4.8). We consider the norms of elements of $E[c]$ over E. Since c has minimal polynomial $t^k - \lambda$ in $E[t]$, it is clear that the norm of c (in $E[c]$, over E) is λ. Since the norm is a multiplicative function, the norm of c^{-1} is λ^{-1}. Finally, since the extensions $E[c]$ and $E[a]$ are isomorphic (as extensions of E), we conclude that there also exists an element d of $E[a]$ such that the norm of d (in $E[a]$, over E) is λ^{-1}, as well.

Proposition 2.5.8. Let A, a, c, and d be as above. Then the element $1 - cd$ of A is a nonzero zero divisor in A.

Proposition 2.5.8 is presented in Herstein [62], and in Rónyai ([102], [103]). It reduces the computation of a nonzero zero divisor in A to the solution of certain "norm equations" in finite fields. Rónyai presents a polynomial time algorithm for the solution of the equation $N_{E[a]/E}(t) = \lambda$, for $\lambda \in E$, over an extension $E[a]$ of dimension k over E, for the case $k = 2$ and for the case k is odd. The general case can be reduced to these special cases: an arbitrary $k \in \mathbb{Z}$ is either a power of two, or has an odd factor greater than one. Given $a \in A$ with (field) $E[a]$ an extension of dimension $k > 1$ over E, it suffices to choose a factor $\hat{k}$ of k which is either 2 or odd. An element $\hat{a}$ of $E[a]$, with $E[\hat{a}]$ a field of dimension $\hat{k}$ over E, is obtained by choosing a polynomial $\hat{f}$ in $E[t]$ which is irreducible in $E[t]$ (randomly, with small probability of failure), and then factoring $\hat{f}$ in $(E[a])[t]$. The element $\hat{a}$ can then be used (in place of the element a) in the procedure CUT; if $\hat{k} \neq k$, then a proper noncommutative subalgebra of A will be returned.

The procedure CUT, and the main algorithm, "ZERODIV", which uses it to generate a zero divisor in A, are stated in more detail by Rónyai ([102], and in a slightly altered form in [103]). Rónyai also shows that the algorithm can be used to compute a nonzero zero divisor of a simple algebra A over $\mathbb{F}_q$ in polynomial time (using a probabilistic algorithm for factorisation of polynomials over $\mathbb{F}_q$). As noted above (and by Rónyai), the decomposition of simple algebras over $\mathbb{F}_q$ can be reduced to this computation.

Theorem 2.5.9. (Rónyai). A simple algebra $A \subseteq M_{m\times m}(\mathbb{F}_q)$ of dimension n over a finite field $\mathbb{F}_q$ can be decomposed using a probabilistic Boolean algorithm, which either successfully decomposes the algebra or returns failure (with probability less than one half), in time $(nm \log q)^{O(1)}$.

Clearly the probability of failure can be made arbitrarily small by repeating the algorithm, with independent choices of random field elements.

Unfortunately, Wedderburn's Theorem, and Propositions 2.5.6, 2.5.7, and 2.5.8 do not generalise well to more general classes of fields; we cannot expect to use Rónyai's algorithm to decompose simple algebras over infinite fields. On the other hand, the use of zero divisors (and arbitrary elements with reducible minimal polynomials) do not require the assumption that the ground field is finite; we will use these techniques when decomposing simple algebras over $\mathbb{C}$ and $\mathbb{R}$.

2.5.3. Simple Algebras over $\mathbb{C}$

We now consider the cost of decomposing simple algebras over $\mathbb{C}$. We assume we are given a simple algebra A over a number field F as input, with the property that $A \otimes_F \mathbb{C}$ is simple over $\mathbb{C}$; we wish to decompose the simple algebra $A \otimes_F \mathbb{C}$. We show that if A has dimension k^2 over F (so that $A \otimes_F \mathbb{C} \cong M_{k\times k}(\mathbb{C})$) then a standard basis for $A \otimes_F \mathbb{C}$ can be computed using a probabilistic Boolean algorithm, in polynomial time (see, in particular, Theorem 2.5.11).

As is the case for algebras over finite fields, the problem of deciding whether a simple algebra over $\mathbb{C}$ is a division algebra is trivial.

Proposition 2.5.10. Let E be an algebraically closed field; then the only finite-dimensional division algebra over E is E itself.

The proof of Proposition 2.5.10 is straightforward. Let d be any element of a finite-dimensional division algebra D over E, and let f be the minimal polynomial of d over E. Since D is a division algebra, f is irreducible in $E[t]$ and, since E is algebraically closed, f is linear. Consequently $d \in E$, as required.

Thus, if we are given a simple algebra A of dimension n over an algebraically closed field (in particular, over $\mathbb{C}$), we can conclude from Proposition 2.5.10 that A is a division algebra if and only if $n = 1$.

Again, we are left with the less trivial problem of exhibiting an algebra isomorphism between a simple algebra $A \otimes_F \mathbb{C}$ of dimension $n = k^2$ over $\mathbb{C}$, and the matrix algebra $M_{k\times k}(\mathbb{C})$. In Section 2.5.2 we noted that the corresponding problem for algebras over finite fields could be reduced to the computation of an idempotent other than 0 or 1 in the algebra. This computation is easy over an algebra $A \otimes_F \mathbb{C}$ over $\mathbb{C}$: We note that since $A \otimes_F \mathbb{C} \cong M_{k\times k}(\mathbb{C})$, $\text{Centre}(A \otimes_F \mathbb{C}) \cong \mathbb{C}$, and we need only consider an arbitrary noncentral element a of A. This element will have a minimal polynomial (over $\mathbb{C}$) with degree at least 2, which will be reducible in $\mathbb{C}[t]$. Thus, an idempotent can be generated from a using the methods discussed in Section 2.5.2. Unfortunately, the reduction used for finite fields is not applicable here: If we use the computation of idempotents as the basis for an iterative decomposition of an algebra $A \otimes_F \mathbb{C}$, then we will generally require the computation of a field extension at each iteration. In the worst case, this simple iterative scheme will produce output with size exponential in the size of the input. We avoid this problem by replacing the iterative algorithm for computation of primitive idempotents by a probabilistic algorithm, which either computes a complete set of primitive idempotents in a single step, or fails.

Suppose again that $A \otimes_F \mathbb{C}$ is simple, with dimension $n = k^2$ over $\mathbb{C}$, so $A \otimes_F \mathbb{C} \cong M_{k\times k}(\mathbb{C})$. Let ϕ be an isomorphism from $A \otimes_F \mathbb{C}$ to $M_{k\times k}(\mathbb{C})$. If $a_1, a_2, \ldots, a_n$ is a basis for A over F, then this is also a basis for $A \otimes_F \mathbb{C}$ over $\mathbb{C}$. For $\lambda_1, \lambda_2, \ldots, \lambda_n \in \mathbb{C}$, we consider the matrix $\phi(\lambda_1 a_1 + \lambda_2 a_2 + \cdots + \lambda_n a_n) \in M_{k\times k}(\mathbb{C})$ (as a function of $\lambda_1, \lambda_2, \ldots, \lambda_n$). We denote by $\chi(\lambda_1, \lambda_2, \ldots, \lambda_n)$ the characteristic polynomial of this matrix; $\chi(\lambda_1, \lambda_2, \ldots, \lambda_n)$ is a polynomial of degree k in $\mathbb{C}[t]$. Now there exist elements $\lambda_1, \lambda_2, \ldots, \lambda_n$ of $\mathbb{C}$ such that

$$\phi(\lambda_1, \lambda_2, \ldots, \lambda_n) = \text{Diag}(1, 2, \ldots, n) = \begin{bmatrix} 1 & & & 0 \\ & 2 & & \\ & & \ddots & \\ 0 & & & n \end{bmatrix},$$

and so $\chi(\lambda_1, \lambda_2, \ldots, \lambda_n)$ is squarefree. Setting

$$h(x_1, x_2, \ldots, x_n) = \text{Res}_t(\chi(x_1, x_2, \ldots, x_n), \frac{\mathrm{d}}{\mathrm{d}t}\chi(x_1, x_2, \ldots, x_n)),$$

we note that h has total degree at most $k(2k-1)$ in $x_1, x_2, \ldots, x_n$, that h is not identically 0, and, for arbitrary $\lambda_1, \lambda_2, \ldots, \lambda_n \in C$, that $h(\lambda_1, \lambda_2, \ldots, \lambda_n) \neq 0$ if and only if the minimal polynomial of the matrix $\chi(\lambda_1 a_1 + \lambda_2 a_2 + \cdots + \lambda_n a_n)$ has degree k and is squarefree in $\mathbb{C}[t]$. We argue, as in Section 2.4.4, that for randomly chosen $\lambda_1, \lambda_2, \ldots, \lambda_n \in F$ (chosen from a finite subset of F), the matrix $\chi(\lambda_1 a_1 + \lambda_2 a_2 + \lambda_n a_n)$ has these properties, and can be used to generate a full set of primitive idempotents in A, with high probability. We use this as the basis for the algorithm stated on the following pages.

Note that we have simplified the definition of the problem for this special case, identifying the division algebra D with the only possible such algebra, $\mathbb{C}$ (and dispensing with the trivial basis $d_1 = 1$ for D over $\mathbb{C}$, in the output).

Algorithm **Decomposition of a Simple Algebra over $\mathbb{C}$**

Input.
- Integers h, n, $m > 0$.
- The coefficients of a monic irreducible polynomial $f \in \mathbb{Z}[t]$
  $f = t^h + \lambda_{h-1}t^{h-1} + \cdots + \lambda_1 t + \lambda_0$ of degree h.
- Coordinates of an isolating rectangle in $\mathbb{C}$ for a root α of f.
- Matrices $a_1, a_2, \ldots, a_n \in M_{m\times m}(F)$, for $F = \mathbb{Q}[\alpha]$, which form the basis for a finite-dimensional simple associative algebra $A \subseteq M_{m\times m}(F)$ of dimension n over F, such that $A \otimes_F \mathbb{C}$ is simple over C.
- Error tolerance $\epsilon > 0$.

Output. EITHER:
- Integer $k > 0$, such that $A \otimes_F \mathbb{C} \cong M_{k\times k}(\mathbb{C})$.
- For integers i and j, such that $1 \le i, j \le k$:
  - Integer $h_{i\,j} > 0$, the coefficients of a monic irreducible polynomial
    $f_{i\,j} = t^{h_{i\,j}} + \lambda_{i,\,j,\,h_{i\,j}-1}t^{h_{i\,j}-1} + \cdots + \lambda_{i,\,j,\,1}t + \lambda_{i,\,j,\,0} \in \mathbb{Z}[t]$
    of degree $h_{i\,j}$, and the coordinates of an isolating rectangle in $\mathbb{C}$, for a root $\alpha_{i\,j}$ of $f_{i\,j}$, such that $F \subseteq E_{i\,j} = \mathbb{Q}[\alpha_{i\,j}] \subseteq \mathbb{C}$.
  - Coefficients of a polynomial
    $g_{i\,j} = \zeta_{i,\,j,\,h_{i\,j}-1}t^{h_{i\,j}-1} + \cdots + \zeta_{i,\,j,\,1}t + \zeta_{i,\,j,\,0} \in \mathbb{Q}[t]$
    such that $\alpha = g_{i\,j}(\alpha_{i\,j})$.
  - Elements $\nu_{i\,j\,g}$ of $E_{i\,j}$, for $1 \le g \le n$, defining an element
    $e_{i\,j} = \sum_{g=1}^{n} \nu_{i\,j\,g}a_g$ of $A \otimes_F \mathbb{C}$, such that the elements $e_{i\,j}$ form a basis for $A \otimes_F \mathbb{C}$ over $\mathbb{C}$, with
    $e_{1\,1} + e_{2\,2} + \cdots + e_{k\,k} = 1$, and $e_{r\,s}e_{t\,u} = \delta_{s\,t}e_{r\,u}$
    for $1 \le r, s, t, u \le k$.

OR: *failure*, with probability less than ϵ.

(1) Set $k = \sqrt{n} \in \mathbb{Z}$.

(2) Choose elements $\lambda_1, \lambda_2, \ldots, \lambda_n$ randomly and independently from a subset I of F, of size $\lceil k(2k-1)\epsilon^{-1} \rceil$.

(3) Compute the (monic) minimal polynomial $\hat{f}$ in $F[t]$ of the element
$a = \lambda_1 a_1 + \lambda_2 a_2 + \cdots + \lambda_n a_n$ of A.
If this polynomial has degree less than k, or has degree k and is not squarefree, report *failure*. Otherwise, perform steps 4–13.

Extraction of Idempotents.

(4) Use factorisation over $\mathbb{C}$ to compute the following values, for $1 \leq i \leq k$.

- The minimal polynomial over $\mathbb{Q}$ (with degree $h_{i\,i}$) and an isolating rectangle in $\mathbb{C}$ for an algebraic integer $\alpha_{i\,i} \in \mathbb{C}$, such that α and the algebraic number β_i both lie in $E_{i\,i} = \mathbb{Q}[\alpha_{i\,i}]$, where β_i is a root of $\hat{f}$ — in particular, so that
$$\hat{f} = (t + \beta_1)(t + \beta_2) \cdots (t + \beta_k)$$
is an irreducible factorisation for $\hat{f}$ over $\mathbb{C}$.
- Elements $a_{i,\,0}$, $a_{i,\,1}$, $\ldots$, $a_{i,\,h_{i\,i}-1}$ and $b_{i,\,0}$, $b_{i,\,1}$, $\ldots$, $b_{i,\,h_{i\,i}-1}$ of $\mathbb{Q}$ such that
$$\alpha = a_{i,\,0} + a_{i,\,1}\alpha_{i\,i} + \cdots + a_{i,\,h_{i\,i}-1}\alpha_{i\,i}^{h_{i\,i}-1},$$
and
$$\beta_i = b_{i,\,0} + b_{i,\,1}\alpha_{i\,i} + \cdots + b_{i,\,h_{i\,i}-1}\alpha_{i\,i}^{h_{i\,i}-1}.$$

Set $g_{i\,i} = a_{i,\,h_{i\,i}-1}t^{h_{i\,i}-1} + \cdots + a_{i,\,1}t + a_{i,\,0} \in \mathbb{Q}[t]$; then $\alpha = g_{i\,i}(\alpha_{i\,i})$.

(5) Compute the polynomial $\hat{g}_i \in E_{i\,i}[t]$ with degree less than k, for $1 \leq i \leq k$, such that
$$\hat{g}_i \equiv 1 \pmod{(t + \beta_i)} \quad \text{and} \quad \hat{g}_i \equiv 0 \pmod{(\hat{f}/(t + \beta_i))}.$$
Set $e_{i\,i}$ to be the idempotent $\hat{g}_i(a) \in A \otimes_F E_i \subseteq A \otimes_F \mathbb{C}$.

Computation of a Standard Basis.

(6) Perform steps 7–10 for all i such that $2 \leq i \leq k$.

(7) Compute the minimal polynomial $f_{1\,i}$ over $\mathbb{Q}$, with degree $h_{1\,i}$, of an algebraic integer $\alpha_{1\,i}$ such that $\mathbb{Q}[\alpha_{1\,i}] = \mathbb{Q}[\alpha_{1\,1}, \alpha_{i\,i}]$, as well as the coordinates for an isolating rectangle for $\alpha_{1\,i}$ in $\mathbb{C}$.
Compute polynomials s_i, $t_i \in E_{1\,i}[t] = (\mathbb{Q}[\alpha_{1\,i}])[t]$ each with degree less than $h_{1\,i}$, such that $\alpha_{1\,1} = s_i(\alpha_{1\,i})$ and $\alpha_{i\,i} = t_i(\alpha_{1\,i})$.
Set $g_{1\,i} = ((g_{1\,1} \circ s_i) \bmod f_{1\,i})$; then $\alpha = g_{1\,i}(\alpha_{1\,i})$.
Set $f_{i\,1} = f_{1\,i}$, $g_{i\,1} = g_{1\,i}$, and $\alpha_{i\,1} = \alpha_{1\,i}$ (so $E_{i\,1} = E_{1\,i} = \mathbb{Q}[\alpha_{1\,i}]$).

(8) Compute a basis over $E_{1\,i}$ for the subspace $(e_{1\,1} + e_{i\,i})(A \otimes_F E_{1\,i})(e_{1\,1} + e_{i\,i})$ of $A \otimes_F E_{1\,i}$, by choosing a maximal linearly independent subset (over $E_{1\,i}$) of $(e_{1\,1} + e_{i\,i})a_1(e_{1\,1} + e_{i\,i})$, $(e_{1\,1} + e_{i\,i})a_2(e_{1\,1} + e_{i\,i})$, $\ldots$, $(e_{1\,1} + e_{i\,i})a_n(e_{1\,1} + e_{i\,i})$.
Note that this is subspace a simple algebra isomorphic to $M_{2\times 2}(E_{1\,i})$ over $E_{1\,i}$ with identity element $e_{1\,1} + e_{i\,i}$.

(9) Use the basis computed in (8) to form and solve a system of linear equations over $E_{1\,i}$, to compute as $e_{1\,i}$ any nonzero element x of the algebra $(e_{1\,1} + e_{i\,i})(A \otimes_F E_{1\,i})(e_{1\,1} + e_{i\,i})$ such that
$$e_{1\,1}x = x, \quad e_{i\,i}x = 0, \quad xe_{1\,1} = 0, \quad \text{and} \quad xe_{i\,i} = x.$$

(10) Form and solve a system of linear equations over $E_{1\,i}$, to compute as $e_{i\,1}$ the unique element y of $(e_{1\,1}+e_{i\,i})(A\otimes_F E_{1\,i})(e_{1\,1}+e_{i\,i})$ satisfying the equations

$$e_{1\,1}y=0, \quad e_{i\,i}y=y, \quad ye_{1\,1}=y, \quad ye_{i\,i}=0,$$
$$ye_{1\,i}=e_{i\,i}, \quad \text{and} \quad e_{1\,i}y=e_{1\,1}.$$

(11) For $2\le i<j\le k$, compute the minimal polynomial $f_{i\,j}$ over $\mathbb{Q}$, with degree $h_{i\,j}$, of an algebraic integer $\alpha_{i\,j}$ such that $\mathbb{Q}[\alpha_{i\,j}]=\mathbb{Q}[\alpha_{1\,i},\alpha_{1\,j}]$, as well as the coordinates of an isolating rectangle for $\alpha_{i\,j}$ in $\mathbb{C}$.
Compute polynomials $s_{i\,j},\, t_{i\,j}\in E_{i\,j}[t]=(\mathbb{Q}[\alpha_{i\,j}])[t]$ with degree less than $h_{i\,j}$ such that $\alpha_{1\,i}=s_{i\,j}(\alpha_{i\,j})$ and $\alpha_{1\,j}=t_{i\,j}(\alpha_{i\,j})$.
Set $g_{i\,j}=((g_{1\,i}\circ s_{i\,j}) \bmod f_{i\,j})$; then $\alpha=g_{i\,j}(\alpha_{i\,j})$.
Set $f_{j\,i}=f_{i\,j}$, $g_{j\,i}=g_{i\,j}$, and $\alpha_{j\,i}=\alpha_{i\,j}$ (so $E_{j\,i}=E_{i\,j}=\mathbb{Q}[\alpha_{i\,j}]$).

(12) For $2\le i,\, j\le k$, with $i\ne j$, set $e_{i\,j}=e_{i\,1}e_{1\,j}\in A\otimes_F E_{i\,j}\subseteq A\otimes_F\mathbb{C}$.

(13) For $1\le i,\, j\le k$, form and solve a nonsingular system of linear equations, to compute the elements $\nu_{i\,j\,g}$ of $E_{i\,j}$ for $1\le g\le n$, such that $e_{i\,j}=\sum_{g=1}^{n}\nu_{i\,j\,g}a_g$ in $A\otimes_F E_{i\,j}\subseteq A\otimes_F\mathbb{C}$.

We are interested in Boolean algorithms for this problem. We assume the error tolerance ϵ is of the form N^{-1} for some integer $N>0$, so that ϵ can be represented using $\Theta(\log_2(\epsilon^{-1}))=\Theta(\log_2 N)$ bits; then the following theorem states that the algorithm requires time polynomial in its input size.

Theorem 2.5.11. Suppose $A\subseteq M_{m\times m}(F)$ is a simple algebra of dimension $n=k^2$ over a number field $F=\mathbb{Q}[\alpha]$ of degree h over $\mathbb{Q}$, for an algebraic integer α, such that $A\otimes_F\mathbb{C}$ is simple over $\mathbb{C}$. Then there exists a standard basis $\{\, e_{i\,j} : 1\le i,\, j\le k\,\}$ for $A\otimes_F\mathbb{C}$ over $\mathbb{C}$ such that $e_{1\,1}+e_{2\,2}+\cdots+e_{k\,k}=1$ and $e_{r\,s}e_{t\,u}=\delta_{s\,t}e_{r\,u}$ for $1\le r,\, s,\, t,\, u\le k$, and such that each element $e_{i\,j}$ lies in a number field $E_{i\,j}$ whose dimension over $\mathbb{Q}$ is polynomial in k and in the dimension of F over $\mathbb{Q}$. Given the minimal polynomial over $\mathbb{Q}$ and an isolating rectangle in $\mathbb{C}$ for α, a basis $a_1,\, a_2,\, \ldots,\, a_n$ for A over F, and an integer $N>0$ (in binary notation) as input, a probabilistic Boolean algorithm can be used to compute a generator $\alpha_{i\,j}$ of a field $E_{i\,j}=\mathbb{Q}[\alpha_{i\,j}]$ and the element $e_{i\,j}$ of $A\otimes_F E_{i\,j}$ such that $\{\, e_{i\,j} : 1\le i,\, j\le k\,\}$ forms a standard basis for $A\otimes_F\mathbb{C}$ as described above, using time polynomial in the input size, with probability of failure less than $\epsilon=N^{-1}$.

Proof. We use the algorithm "Decomposition of a Simple Algebra over $\mathbb{C}$" to perform this computation. The techniques used here to generate a set of primitive idempotents from an element of A whose minimal polynomial is squarefree and has (full) degree k (in steps 4–5), and to obtain a standard basis from these idempotents (in steps 6–13), have been discussed before, and have been shown to be correct. We will show that the bound on the probability of error is correct. We will then show that the number fields $E_{i\,j}$ used by the algorithm have small dimension over F, and conclude (citing results from Section 1 and in the article of Loos [87]) that the algorithm computes the desired values in polynomial time.

We have already noted that an arbitrary element of A has a minimal polynomial (over F) with degree at most k, and that there exists an element whose minimal polynomial is squarefree and has degree k. We note, as in Section 2.4, that there exists a polynomial $h(x_1, x_2, \ldots, x_n)$ with total degree at most $k(2k-1)$, such that for $\lambda_1, \lambda_2, \ldots, \lambda_n \in F$, $h(\lambda_1, \lambda_2, \ldots, \lambda_n) \neq 0$ if and only if the element $\lambda_1 a_1 + \lambda_2 a_2 + \cdots + \lambda_n a_n$ of A has a minimal polynomial over F which is squarefree and of degree k, so that this element can be used to generate a set of primitive idempotents which can serve as the elements $e_{1\,1}, e_{2\,2}, \ldots, e_{k\,k}$ of a standard basis. Applying the result of Schwartz [111](which we stated as Proposition 2.4.23), we conclude that if $\lambda_1, \lambda_2, \ldots, \lambda_n$ are chosen randomly from a finite subset I of F with size $\lceil k(2k-1)\epsilon^{-1} \rceil$ in step 2 of the algorithm, then the algorithm will fail in step 3 with probability at most ϵ, as desired.

We now note that the number fields $E_{i\,j}$ have small dimension over F. Since $E_{i\,i}$ is obtained from F by adjoining a single root of a polynomial $\hat{f} \in F[t]$ with degree k, it is clear that $[E_{i\,i} : F] \leq k$ for $1 \leq i \leq k$. The field $E_{1\,i}$ is obtained from $E_{1\,1}$ by adjoining an element $(\alpha_{i\,i})$ of $E_{i\,i}$; since $E_{1\,1}$ and $E_{i\,i}$ each have dimension at most k over F, it follows that $E_{1\,i}$ has dimension at most k^2 over F. Finally, $E_{i\,j}$ is obtained from $E_{1\,1}$ by adjoining elements from $E_{i\,i}$ and from $E_{j\,j}$; so $E_{i\,j}$ has dimension at most k^3 over F, for arbitrary i and j. Thus these fields each have dimension at most $k^3 l$ over $\mathbb{Q}$; generators of these fields over $\mathbb{Q}$ can be obtained (by the algorithm of Loos) in polynomial time, and arithmetic can also be performed efficiently over these fields (by the methods of Section 1). The remaining analysis is straightforward. We conclude that the algorithm can be implemented to run in time polynomial in the input size, as claimed. ■

As indicated above, we obtain a polynomial-size standard basis for the algebra $A \otimes_F \mathbb{C}$ using the algorithm "Decomposition of a Simple Algebra over $\mathbb{C}$" — but only by choosing each entry $e_{i\,j}$ from an algebra $A \otimes_F E_{i\,j}$, where $E_{i\,j}$ is an extension of F particular to that entry of the basis. Note that the algorithm is complicated by the need to form and keep track of all of these field extensions. A standard basis $\{\, e_{i\,j} : 1 \leq i, j \leq n \,\}$ with $e_{i\,j} \in A \otimes_F E$, for a *single* extension E with small degree over F, would be preferable to the one we obtain.

Babai and Rónyai [5] have subsequently improved the results presented here, by showing that such a single extension E can be found. In particular, their results can be used to show that if steps 1–4 of "Decomposition of a Simple Algebra over $\mathbb{C}$" are performed, and if the method described in step 5 is used to obtain a *single* idempotent $e_{1\,1}$ in $A \otimes_F E$, so that $(A \otimes_F E)e_{1\,1}$ is an irreducible left ideal in

$A \otimes_F E$, then in fact $A \otimes_F E$ is isomorphic to a full matrix ring over E, and the rest of a standard basis for $A \otimes_F E$ over E can be computed in polynomial time. (Note that the centre of A is isomorphic to F, so that Lemma 2.5 of [5] applies.) Now a_1, a_2, $\ldots$, a_n spans A over F, and hence also spans $A \otimes_F E$ over E. For $1 \leq i \leq n$, let $\hat{L}_i$ be the left ideal of $A \otimes_F E$ generated by the elements $e_{1\,1}a_1$, $e_{1\,1}a_2$, $\ldots$, $e_{1\,1}a_i$. It is clear that

$$\hat{L}_1 \subseteq \hat{L}_2 \subseteq \cdots \subseteq \hat{L}_n,$$

and that this set of ideals includes an ideal of dimension ik over n, for $1 \leq i \leq k$. Set $\hat{e}_i$ to be a maximal idempotent in $\hat{L}_i$, for $1 \leq i \leq n$, and set $\tilde{e}_1 = \hat{e}_1$, $\tilde{e}_{i+1} = (1-\hat{e}_i)\hat{e}_{i+1}$ for $1 \leq i < n$. Eliminating the idempotents equal to zero, we obtain from the $\tilde{e}_i$'s a set of primitive idempotents in $A \otimes_F E$. The algorithm "Decomposition from Primitive Idempotents" of Section 2.5.1 can then be used (over the field E) to compute a standard basis for $A \otimes_F E$ over E from these idempotents.

We have seen that the problem of identifying a division algebra over E is trivial when E is algebraically closed (in particular, when $E = \mathbb{C}$). We used a probabilistic method in order to avoid the use of iteration (and a large increase in the size of the values being computed) when producing an isomorphism between a simple algebra over $\mathbb{C}$ and a matrix ring $M_{k \times k}(\mathbb{C})$. By doing so, we have shown that a "small" (i.e., polynomial size) standard basis exists for a simple algebra $A \otimes_F \mathbb{C}$, for a simple algebra A over a number field F. The question of whether such a basis can be computed deterministically in polynomial time remains open.

Question 2.5.12. Given a basis for an algebra $A \subseteq M_{m \times m}(F)$ over a number field F, with $A \otimes_F \mathbb{C}$ simple over $\mathbb{C}$, can an isomorphism between $A \otimes_F \mathbb{C}$ and $M_{k \times k}(\mathbb{C})$ be exhibited (by computing a standard basis for $A \otimes_F \mathbb{C}$) using a deterministic Boolean algorithm, in time polynomial in the input size?

Babai and Rónyai have answered this question for an important special case: They obtain such a decomposition for a group algebra over a splitting field for that algebra (in particular, over any extension of $\mathbb{Q}$ containing an e^{th} primitive root of unity, for e the exponent of the associated group).

2.5.4. Simple Algebras over $\mathbb{R}$

The decomposition of simple algebras over $\mathbb{R}$ is a more complicated process than the decomposition of simple algebras over $\mathbb{C}$. In particular, the decision problem of deciding whether a simple algebra over $\mathbb{R}$ is a division algebra over $\mathbb{R}$ is nontrivial; for we have seen that a noncommutative finite-dimensional division algebra over $\mathbb{R}$ exists (namely, $\mathbb{H}$, the ring of real quaternions, of Examples 2.1.3 and 2.2.4). Fortunately, the number of distinct finite-dimensional division algebras over $\mathbb{R}$ is small.

Proposition 2.5.12. The only finite-dimensional division algebras over $\mathbb{R}$ are $\mathbb{R}$ itself, $\mathbb{C} = \mathbb{R}[\sqrt{-1}]$, and $\mathbb{H}$, the ring of real quaternions.

Proposition 2.5.12 is well known; for a proof see van der Waerden ([118], Section 14.9). We will use Proposition 2.5.12 to decompose simple algebras, by producing a probabilistic Boolean algorithm which identifies, and decomposes, algebras isomorphic to $M_{k\times k}(D)$, for each of the three cases $D = \mathbb{R}$, $D = \mathbb{C}$, and $D = \mathbb{H}$. Given an algebra $\hat{A} = A \otimes_F \mathbb{R}$, for $F = \mathbb{Q}[\alpha] \subseteq \mathbb{R}$ a number field, such that $\hat{A}$ is simple over $\mathbb{R}$, the algorithm computes a standard basis $\{\, e_{i\,j} \;:\; 1 \le i,\, j \le k \,\}$ for $\hat{A} \cong M_{k\times k}(D)$ over D, with each element $e_{i\,j} \in A \otimes_F E_{i\,j}$, for $F \subseteq E_{i\,j} \subseteq \mathbb{R}$, with $E_{i\,j}$ a finite algebraic extension with dimension $k^{O(1)}$ over F, as well as the dimension l of D over F, and elements $d_1e_{i\,j}$, $d_2e_{i\,j}$, $\ldots$, $d_le_{i\,j} \in A \otimes_F E_{i\,j}$, where $d_1, d_2, \ldots, d_l \in A \otimes_F \mathbb{R}$ form a basis for a division algebra isomorphic to D over $\mathbb{R}$. To our knowledge, this is the first (probabilistic) polynomial-time algorithm for this problem. We discuss a subsequent improvement of our algorithm at the end of this section.

We first note that simple algebras isomorphic to $M_{k\times k}(\mathbb{C})$ over $\mathbb{R}$ are easily distinguished from other simple algebras over $\mathbb{R}$.

Proposition 2.5.13. Let $\hat{A}$ be a simple algebra over $\mathbb{R}$; then $\hat{A} \cong M_{k\times k}(\mathbb{C})$ for some $k \ge 1$ if and only if Centre($\hat{A}$) has dimension two over $\mathbb{R}$ (and is isomorphic to $\mathbb{C}$). Otherwise, $\hat{A} \cong M_{k\times k}(\mathbb{R})$ or $\hat{A} \cong M_{k\times k}(\mathbb{H})$ for some $k > 0$, and Centre($\hat{A}$) $\cong \mathbb{R}$.

Proposition 2.5.13 follows easily from the fact that Centre($M_{k\times k}(D)$) $\cong$ Centre(D) over $\mathbb{R}$, and that Centre($\mathbb{C}$) $\cong \mathbb{C}$, while Centre($\mathbb{H}$) $\cong$ Centre($\mathbb{R}$) $\cong \mathbb{R}$.

Suppose now that $\hat{A} = A \otimes_F \mathbb{R}$ and Centre($\hat{A}$) $\cong \mathbb{C}$; consider a basis $\{\, \alpha_1 = 1,\, \alpha_2 \,\}$ for Centre($\hat{A}$) over $\mathbb{R}$. We first note that α_2 must have a quadratic minimal polynomial $t^2+\gamma_1 t+\gamma_0$ over $\mathbb{R}$. Since this polynomial is irreducible over $\mathbb{R}$, $\gamma_1^2-4\gamma_0 < 0$; hence $\sqrt{4\gamma_0-\gamma_1^2} \in \mathbb{R}$ and $\sqrt{4\gamma_0-\gamma_1^2} \ne 0$. Let $I = (4\gamma_0-\gamma_1^2)^{-(1/2)}(2\alpha_2+\gamma_1)$; then it is easily checked that $I \in$ Centre($A \otimes_F E$) and that I is a root of the polynomial t^2+1, for $E = F[\sqrt{4\gamma_0-\gamma_1^2}]$. Now $\{\, 1,\, I \,\}$ is a basis for Centre($A \otimes_F E$) over E, with $I^2+1=0$. We decompose $A \otimes_F \mathbb{R} = (A \otimes_F E) \otimes_E \mathbb{R}$ by computing a basis, $\beta_1, \beta_2, \ldots, \beta_{\hat{n}}$, for $A \otimes_F E$ over Centre$(A \otimes_F E) \cong E[\sqrt{-1}]$. (Note that $\beta_1, \beta_1 I, \beta_2, \beta_2 I, \ldots, \beta_{\hat{n}}, \beta_{\hat{n}} I$ is then a basis for $A \otimes_F E$ over E.) We use the basis $\beta_1, \beta_2, \ldots, \beta_{\hat{n}}$ (of size $\hat{n} = n/2 = k^2$ for $k \in \mathbb{Z}$) to compute a set of structure constants for $A \otimes_F E$ (now viewed as an associative algebra $\bar{A}$ over Centre$(A \otimes_F E) \cong E[\sqrt{-1}]$), which we use as input for the algorithm "Decomposition of a Simple Algebra over $\mathbb{C}$" of Section 2.5.3.

The above algorithm will return a set of number fields $E_{i\,j} \subseteq \mathbb{C}$ and a standard basis $\{\, e_{i\,j} \;:\; 1 \leq i, j \leq k \,\}$ for the algebra $\bar{A} \otimes_{E[\sqrt{-1}]} \mathbb{C}$ over $\mathbb{C}$. Each element $e_{i\,j}$ will be given by a set of elements $\nu_{i\,j\,g} \in E_{i\,j}$, for $1 \leq g \leq \hat{n}$, such that

$$e_{i\,j} = \sum_{g=1}^{\hat{n}} \nu_{i\,j\,g} \beta_g \in \bar{A} \otimes_{E[\sqrt{-1}]} E_{i\,j}.$$

Now the field $E_{i\,j}$ is given as $\mathbb{Q}[\alpha_{i\,j}]$, for some algebraic integer $\alpha_{i\,j}$; using the algorithms of Loos [87], we can compute a generator $\hat{\alpha}_{i\,j}$ for the number field $\hat{E}_{i\,j} = \mathbb{Q}[\hat{\alpha}_{i\,j}] = \mathbb{Q}[\mathrm{Re}(\alpha_{i\,j}), \mathrm{Im}(\alpha_{i\,j})] \subseteq \mathbb{R}$. It is clear that $\hat{E}_{i\,j} = (E_{i\,j}[\sqrt{-1}]) \cap \mathbb{R}$; since $\nu_{i\,j\,g} \in E_{i\,j}$, $\mathrm{Re}(\nu_{i\,j\,g})$, $\mathrm{Im}(\nu_{i\,j\,g}) \in \hat{E}_{i\,j}$ for $1 \leq g \leq \hat{n}$. Now let

$$\hat{e}_{i\,j} = \sum_{g=1}^{\hat{n}} (\mathrm{Re}(\nu_{i\,j\,g})\beta_g + \mathrm{Im}(\nu_{i\,j\,g})\beta_g I) \in A \otimes_F \hat{E}_{i\,j} \subseteq A \otimes_F \mathbb{R}.$$

It is clear that the elements $\hat{e}_{i\,j}$ form a standard basis for $A \otimes_F \mathbb{R}$ over the field Centre$(A \otimes_F \mathbb{R}) \cong \mathbb{C}$, and that $d_1 = 1$ and $d_2 = I$ form a basis for this field over $\mathbb{R}$, yielding a decomposition of $A \otimes_F \mathbb{R}$ over $\mathbb{R}$. It is also easily checked that these values can be recovered from the values returned by the algorithm "Decomposition of a Simple Algebra over $\mathbb{C}$" in polynomial time. As we note at the end of Section 2.5.3, we can use the methods of Babai and Rónyai [5] to improve this result: We can choose all the elements $\hat{e}_{i\,j}$ and $I\hat{e}_{i\,j}$ from $A \otimes_F E$, where $E \subseteq \mathbb{R}$ and E is a *single* extension of F, of small degree over F.

Henceforth we assume Centre$(\hat{A}) \cong \mathbb{R}$, so either $\hat{A} \cong M_{k \times k}(\mathbb{R})$ or $\hat{A} \cong M_{k \times k}(\mathbb{H})$ for some $k > 0$. If $\hat{A}$ has dimension n over $\mathbb{R}$ and $\hat{A} \cong M_{k \times k}(\mathbb{R})$, then $n = k^2$; otherwise, $\hat{A} \cong M_{k \times k}(\mathbb{H})$, and $n = 4k^2$. We can conclude immediately that $\hat{A} \cong M_{k \times k}(\mathbb{R})$ if n is odd. However, we are still left with the problem of decomposing the algebra by computing a standard basis, as well as the problem of classifying and decomposing simple algebras of even dimension over $\mathbb{R}$. In the rest of this section we develop a procedure which solves these problems by reducing them to the special case that $\hat{A}$ has dimension 4 over $\mathbb{R}$, and either $\hat{A} \cong \mathbb{H}$ or $\hat{A} \cong M_{2 \times 2}(\mathbb{R})$ — that is, to the case that $\hat{A}$ is a *quaternion algebra* over $\mathbb{R}$. We also give a procedure which solves these problems for quaternion algebras over $\mathbb{R}$ in polynomial time.

Suppose now that $\hat{A} = A \otimes_F \mathbb{R}$ has dimension $n = \hat{k}^2$ over $\mathbb{R}$, for $\hat{k}$ a positive integer. Suppose $\hat{A} \cong M_{\hat{k}\times\hat{k}}(\mathbb{R})$; then it is clear that an arbitrary element a of $\hat{A}$ has a minimal polynomial over $\mathbb{R}$ with degree at most $\hat{k}$. Suppose ϕ is an isomorphism from $\hat{A}$ to $M_{\hat{k}\times\hat{k}}(\mathbb{R})$; then there exists some element a of $\hat{A}$ such that

$$\phi(a) = \text{Diag}(1,\, 2,\, \ldots,\, \hat{k}) = \begin{bmatrix} 1 & & & 0 \\ & 2 & & \\ & & \ddots & \\ 0 & & & \hat{k} \end{bmatrix},$$

and whose minimal polynomial over $\mathbb{R}$ is squarefree and has degree (exactly) $\hat{k}$. We now note that these facts about the minimal polynomial over $\mathbb{R}$ also hold for elements of $M_{k\times k}(\mathbb{H})$, for $\hat{k} = 2k$.

Lemma 2.5.14. Suppose $\hat{A} \cong M_{k\times k}(\mathbb{H})$ over $\mathbb{R}$, for $k > 0$.

(i) Every element a of $\hat{A}$ has a minimal polynomial (in $\mathbb{R}[t]$) over $\mathbb{R}$ with degree at most $2k$.

(ii) There exists an element a of $\hat{A}$ whose minimal polynomial over $\mathbb{R}$ is squarefree and has degree $2k$.

Proof. To prove (i), we recall that the division algebra $\mathbb{H}$ has a basis $\{\, 1,\, i,\, j,\, k \,\}$ over $\mathbb{R}$, with multiplicative identity 1, and such that $i^2 = j^2 = k^2 = -1$, $ij = -ji = k$, $jk = -kj = i$, and $ki = -ik = j$. Now there is a well known injective homomorphism $\phi : \mathbb{H} \to M_{2\times 2}(\mathbb{C})$ with

$$\phi(1) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \qquad \phi(i) = \begin{bmatrix} \sqrt{-1} & 0 \\ 0 & -\sqrt{-1} \end{bmatrix},$$

$$\phi(j) = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}, \qquad \phi(k) = \begin{bmatrix} 0 & \sqrt{-1} \\ \sqrt{-1} & 0 \end{bmatrix}.$$

(See, for example, the discussion of quaternions in [67]). This can be extended to obtain an injective homomorphism $\hat{\phi} : \hat{A} \to M_{2k\times 2k}(\mathbb{C})$, with

$$\hat{\phi}(e_{r\,s})_{t\,u} = \begin{cases} 1, & \text{if } t = 2r-1 \text{ and } u = 2s-1, \\ 1, & \text{if } t = 2r \text{ and } u = 2s, \\ 0, & \text{otherwise,} \end{cases}$$

for $1 \le r,\, s \le k$ and $1 \le t,\, u \le 2k$ (and for $\{\, e_{r\,s} \;:\; 1 \le r,\, s \le k \,\}$ a "standard basis" for $\hat{A} \cong M_{k\times k}(\mathbb{H})$ over $\mathbb{H}$), and such that $\hat{\phi}(1)$, $\hat{\phi}(i)$, $\hat{\phi}(j)$, and $\hat{\phi}(k)$ are all block diagonal matrices, with 2×2 blocks, and with

$$\hat{\phi}(1) = \begin{bmatrix} \phi(1) & & & 0 \\ & \phi(1) & & \\ & & \ddots & \\ 0 & & & \phi(1) \end{bmatrix}, \qquad \hat{\phi}(i) = \begin{bmatrix} \phi(i) & & & 0 \\ & \phi(i) & & \\ & & \ddots & \\ 0 & & & \phi(i) \end{bmatrix},$$

$$\hat{\phi}(j) = \begin{bmatrix} \phi(j) & & & 0 \\ & \phi(j) & & \\ & & \ddots & \\ 0 & & & \phi(j) \end{bmatrix}, \qquad \hat{\phi}(k) = \begin{bmatrix} \phi(k) & & & 0 \\ & \phi(k) & & \\ & & \ddots & \\ 0 & & & \phi(k) \end{bmatrix}.$$

We state without proof that $\hat{\phi}$ is an injective homomorphism of associative algebras over $\mathbb{R}$.

Now it follows that, for arbitrary $a \in M_{k\times k}(\mathbb{H})$, $\hat{\phi}(a)$ is a $2k \times 2k$ matrix over $\mathbb{C}$ of the form

$$\hat{\phi}(a) = \begin{bmatrix} A_{1\,1} & A_{1\,2} & \cdots & A_{1\,k} \\ A_{2\,1} & A_{2\,2} & \cdots & A_{2\,k} \\ \vdots & \vdots & \ddots & \vdots \\ A_{k\,1} & A_{k\,2} & \cdots & A_{k\,k} \end{bmatrix},$$

where each $A_{i\,j}$ is a 2×2 matrix over $\mathbb{C}$ of the form

$$A_{i\,j} = \begin{bmatrix} a_{i\,j} + b_{i\,j}\sqrt{-1} & c_{i\,j} + d_{i\,j}\sqrt{-1} \\ -c_{i\,j} + d_{i\,j}\sqrt{-1} & a_{i\,j} - b_{i\,j}\sqrt{-1} \end{bmatrix}, \qquad \text{for } a_{i\,j},\, b_{i\,j},\, c_{i\,j},\, d_{i\,j} \in \mathbb{R},$$

and for $1 \le i,\, j \le k$. Let χ_a be the characteristic polynomial of the (complex) matrix $\hat{\phi}(a)$; then χ_a has degree $2k$ over $\mathbb{C}$ and, since $\hat{\phi}$ is an injective homomorphism, $\chi_a(a) = 0$ in $A \otimes_F \mathbb{C}$. We prove part (i) by showing that χ_a has real coefficients, so that the minimal polynomial of a over $\mathbb{R}$ is a divisor of χ_a, and has degree at most $2k$.

Now $\chi_a = \det T \in \mathbb{C}[t]$ for

$$T = \begin{bmatrix} t - A_{1\,1} & -A_{1\,2} & \cdots & -A_{1\,k} \\ -A_{2\,1} & t - A_{2\,2} & \cdots & -A_{2\,k} \\ \vdots & \vdots & \ddots & \vdots \\ -A_{k\,1} & -A_{k\,2} & \cdots & t - A_{k\,k} \end{bmatrix} \in M_{2k\times 2k}(\mathbb{C}[t]);$$

let

$$U = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}, \qquad V = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix},$$

so $\det U = \det V = 1$, and $UV = 1$, and let $\hat{U}$, $\hat{V} \in M_{2k\times 2k}(\mathbb{C})$ with

$$\hat{U} = \begin{bmatrix} U & & & 0 \\ & U & & \\ & & \ddots & \\ 0 & & & U \end{bmatrix}, \qquad \hat{V} = \begin{bmatrix} V & & & 0 \\ & V & & \\ & & \ddots & \\ 0 & & & V \end{bmatrix}.$$

Again, det $\hat{U} = \det \hat{V} = 1$, so

$$\begin{aligned}
\chi_a &= \det\left(\hat{U}\begin{bmatrix} t - A_{1\,1} & -A_{1\,2} & \cdots & -A_{1\,k} \\ -A_{2\,1} & t - A_{2\,2} & \cdots & -A_{2\,k} \\ \vdots & \vdots & \ddots & \vdots \\ -A_{k\,1} & -A_{k\,2} & \cdots & t - A_{k\,k} \end{bmatrix}\hat{V}\right) \\
&= \det\begin{bmatrix} t - \bar{A}_{1\,1} & -\bar{A}_{1\,2} & \cdots & -\bar{A}_{1\,k} \\ -\bar{A}_{2\,1} & t - \bar{A}_{2\,2} & \cdots & -\bar{A}_{2\,k} \\ \vdots & \vdots & \ddots & \vdots \\ -\bar{A}_{k\,1} & -\bar{A}_{k\,2} & \cdots & t - \bar{A}_{k\,k} \end{bmatrix},
\end{aligned}$$

for

$$\begin{aligned}
\bar{A}_{i\,j} &= U\begin{bmatrix} a_{i\,j} + b_{i\,j}\sqrt{-1} & c_{i\,j} + d_{i\,j}\sqrt{-1} \\ -c_{i\,j} + d_{i\,j}\sqrt{-1} & a_{i\,j} - b_{i\,j}\sqrt{-1} \end{bmatrix} V \\
&= \begin{bmatrix} a_{i\,j} - b_{i\,j}\sqrt{-1} & c_{i\,j} - d_{i\,j}\sqrt{-1} \\ -c_{i\,j} - d_{i\,j}\sqrt{-1} & a_{i\,j} + b_{i\,j}\sqrt{-1} \end{bmatrix}.
\end{aligned}$$

Thus $\chi_a = \det T = \det \bar{T}$, for $\bar{T}$ the matrix whose entries are the complex conjugates of the entries of T. It follows that the coefficients of χ_a are their own complex conjugates. Thus $\chi_a \in \mathbb{R}[t]$, as required to prove part (i).

To prove part (ii), it suffices to note that there exists an element a of $\hat{A}$ such that

$$\hat{\phi}(a) = \begin{bmatrix} 0 & 1 & & & & & & 0 \\ -1 & 0 & & & & & & \\ & & 0 & 2 & & & & \\ & & -2 & 0 & & & & \\ & & & & \ddots & & & \\ & & & & & & 0 & k \\ 0 & & & & & & -k & 0 \end{bmatrix},$$

and such that $\chi_a = (t^2+1^2)(t^2+2^2)\cdots(t^2+k^2)$, so that this polynomial is squarefree in $\mathbb{R}[t]$. It follows that the minimal polynomial of a over $\mathbb{R}$ is χ_a, and is squarefree of degree $2k$, as desired. ■

Suppose $a_1, a_2, \ldots, a_n$ is a basis for A over F, for $n = 4k^2$, and with the algebra $A \otimes_F \mathbb{R}$ isomorphic to one of the matrix algebras $M_{2k\times 2k}(\mathbb{R})$ or $M_{k\times k}(\mathbb{H})$. Then we can use Lemma 2.5.14 and the preceding remarks about $M_{2k\times 2k}(\mathbb{R})$, and argue as in Section 2.5.3, to show that there exists a polynomial h in indeterminates $x_1, x_2, \ldots, x_n$ over F, with total degree at most $2k(4k-1)$, such that for arbitrary $\lambda_1, \lambda_2, \ldots, \lambda_n \in F$, the element

$$a = \lambda_1 a_1 + \lambda_2 a_2 + \cdots + \lambda_n a_n \in A \subseteq A \otimes_F \mathbb{R}$$

has a minimal polynomial (over $\mathbb{R}$) which has degree $2k$ and which is squarefree, if and only if $h(\lambda_1, \lambda_2, \ldots, \lambda_n) \neq 0$. Applying the result of Schwartz (Proposition 2.4.23), we conclude that if $\epsilon > 0$ and I is a finite subset of F such that $|I| \geq \lceil 2k(4k-1)\epsilon^{-1} \rceil$, and if $\lambda_1, \lambda_2, \ldots, \lambda_n$ are chosen randomly and independently from I, then the above element a will have a minimal polynomial $\hat{f}$ (over $\mathbb{R}$) which is squarefree and of degree $2k$, with probability at least $1 - \epsilon$.

If $\hat{A} \cong M_{k\times k}(\mathbb{H})$, then, since the minimal polynomial $\hat{f}$ can have at most k irreducible factors in $\mathbb{R}[t]$, it will follow that

$$\hat{f} = \hat{f}_1 \hat{f}_2 \cdots \hat{f}_k,$$

for distinct, monic, irreducible quadratic polynomials $\hat{f}_1, \hat{f}_2, \ldots, \hat{f}_k \in \mathbb{R}[t]$. The element a can then be used to generate a complete set of primitive idempotents $e_1, e_2, \ldots, e_k$ in $\hat{A}$, in this case. Unfortunately, this is not the case if $\hat{A} \cong M_{2k\times 2k}(\mathbb{R})$; while $\hat{f}$ will (probably) be squarefree and have degree $2k$, we have no guarantee that $\hat{f}$ will split into linear factors over $\mathbb{R}$. In the worst case, $\hat{f}$ will split into k irreducible factors, each of degree two. In general, $\hat{f}$ will have h factors, for $k \leq h \leq 2k$, with some factors linear and some quadratic. The polynomial $\hat{f}$ and the element a can then be used to generate a set of idempotents $e_1, e_2, \ldots, e_h$ such that $e_1 + e_2 + \cdots + e_h = 1$ and $e_r e_s = \delta_{r\,s} e_r$ for $1 \leq r,\, s \leq h$, and such that for $1 \leq i \leq h$, either $e_i \hat{A} e_i \cong \mathbb{R}$ or $e_i \hat{A} e_i \cong M_{2\times 2}(\mathbb{R})$ (by Proposition 2.5.1). We are left with the problem of distinguishing between (and decomposing) algebras isomorphic to $\mathbb{H}$ and algebras isomorphic to $M_{2\times 2}(\mathbb{R})$.

We now consider the problem of decomposing these "leftover" algebras. We will decompose an algebra $\hat{A}$ isomorphic to one of $\mathbb{H}$ or $M_{2\times 2}(\mathbb{R})$ by generating one of two bases for $\hat{A}$: a standard basis $\{ e_{1\,1}, e_{1\,2}, e_{2\,1}, e_{2\,2}, \}$ if $\hat{A} \cong M_{2\times 2}(\mathbb{R})$, or a basis $\{ 1, i, j, k \}$ (with the multiplication table given earlier) if $A \cong \mathbb{H}$. We assume we start with a basis $\{ a_1, a_2, a_3, a_4 \}$ for $\hat{A}$. Since we can find the multiplicative identity of $\hat{A}$ by solving a system of linear equations in 4 unknowns, we assume without loss of generality that $a_1 = 1$.

We decompose $\hat{A}$ by assuming $\hat{A} \cong \mathbb{H}$, and attempting to find a basis of the form given above until we either succeed, or find a nonzero zero divisor in $\hat{A}$ (which we can use to generate idempotents, and a standard basis for $\hat{A} \cong M_{2\times 2}(\mathbb{R})$). Suppose now that a_2 has minimal polynomial $t^2 + \gamma_1 t + \gamma_0$ over $\mathbb{R}$ (a_2 is not a scalar multiple of $1 = a_1$, so the minimal polynomial of a_2 must be quadratic). If the discriminant $\gamma_1^2 - 4\gamma_0$ is nonnegative, then the minimal polynomial is reducible over $\mathbb{R}$, so that $\hat{A} \cong M_{2\times 2}(\mathbb{R})$ and the element a_2 can be used to generate a standard basis. Otherwise, $t^2 + \gamma_1 t + \gamma_0$ is irreducible, $4\gamma_0 - \gamma_1^2 > 0$, and we can compute an element $I = (4\gamma_0 - \gamma_1^2)^{-(1/2)}(2a_2 + \gamma_1)$ in $\hat{A}$ such that $I^2 + 1 = 0$. We next verify that the element I can be used as "i" in the basis to be generated if $\hat{A} \cong \mathbb{H}$.

Lemma 2.5.15. Let I be any element of $\mathbb{H}$ such that $I^2 + 1 = 0$. Then there exist elements J and K of $\mathbb{H}$ such that 1, I, J, K is a basis for $\mathbb{H}$ over $\mathbb{R}$, and such that $I^2 = J^2 = K^2 = -1$, $IJ = -JI = K$, $JK = -KJ = I$, and $KI = -IK = J$.

Proof. We know that there exists a basis 1, i, j, k for $\mathbb{H}$ over $\mathbb{R}$ with the multiplication table indicated above. Let α_0, α_1, α_2, $\alpha_3 \in \mathbb{R}$ such that

$$I = \alpha_0 1 + \alpha_1 i + \alpha_2 j + \alpha_3 k.$$

Using the condition $(\alpha_0 1 + \alpha_1 i + \alpha_2 j + \alpha_3 k)^2 = -1$, we conclude that $\alpha_0 = 0$ and that $\alpha_1^2 + \alpha_2^2 + \alpha_3^2 = 1$. Now there exist elements β_1, β_2, $\beta_3 \in \mathbb{R}$ such that $\beta_1^2 + \beta_2^2 + \beta_3^2 = 1$ and $\alpha_1\beta_1 + \alpha_2\beta_2 + \alpha_3\beta_3 = 0$. (This is equivalent to the condition that the vectors $(\alpha_1, \alpha_2, \alpha_3)$ and $(\beta_1, \beta_2, \beta_3)$ be orthogonal and of unit length with respect to the standard inner product for $\mathbb{R}^3$.) Setting

$$J = \beta_1 i + \beta_2 j + \beta_3 k,$$

we see that $J^2 + 1 = 0$ (since $\beta_1^2 + \beta_2^2 + \beta_3^2 = 1$). Setting $K = IJ$, we see that

$$\begin{aligned} K &= -(\alpha_1\beta_1 + \alpha_2\beta_2 + \alpha_3\beta_3) + (\alpha_2\beta_3 - \alpha_3\beta_2)i \\ &\quad + (\alpha_3\beta_1 - \alpha_1\beta_3)j + (\alpha_1\beta_2 - \alpha_2\beta_1)k \\ &= (\alpha_2\beta_3 - \alpha_3\beta_2)i + (\alpha_3\beta_1 - \alpha_1\beta_3)j \\ &\quad + (\alpha_1\beta_2 - \alpha_2\beta_1)k, \end{aligned}$$

since $\alpha_1\beta_1 + \alpha_2\beta_2 + \alpha_3\beta_3 = 0$. Now we note that for an arbitrary element of $\mathbb{H}$ of the form $\gamma_1 i + \gamma_2 j + \gamma_3 k$, for γ_1, γ_2, $\gamma_3 \in \mathbb{R}$, we have

$$\begin{aligned} (\gamma_1 i + \gamma_2 j + \gamma_3 k)^2 &= -(\gamma_1^2 + \gamma_2^2 + \gamma_3^2) \\ &\quad + (\gamma_2\gamma_3 - \gamma_3\gamma_2)i \\ &\quad + (\gamma_3\gamma_1 - \gamma_1\gamma_3)j \\ &\quad + (\gamma_1\gamma_2 - \gamma_2\gamma_1)k \\ &= -(\gamma_1^2 + \gamma_2^2 + \gamma_3^2), \end{aligned}$$

so $(\gamma_1 i + \gamma_2 j + \gamma_3 k)^2 \in \mathbb{R}$ and $(\gamma_1 i + \gamma_2 j + \gamma_3 k)^2 < 0$. In particular, $K^2 < 0$, and

$$\begin{aligned} K^2 &= -[(-\alpha_2\beta_3)^2 + (\alpha_3\beta_1)^2 + (\alpha_1\beta_2 - \alpha_2\beta_1)^2] \\ &= -\alpha_1^2(\beta_2^2 + \beta_3^2) - \alpha_2^2(\beta_1^2 + \beta_3^2) - \alpha_3^2(\beta_1^2 + \beta_2^2) \\ &\quad + 2\alpha_1\alpha_2\beta_1\beta_2 + 2\alpha_1\alpha_3\beta_1\beta_3 + 2\alpha_2\alpha_3\beta_2\beta_3 \\ &= -(\alpha_1^2 + \alpha_2^2 + \alpha_3^2)(\beta_1^2 + \beta_2^2 + \beta_3^2) + (\alpha_1\beta_1 + \alpha_2\beta_2 + \alpha_3\beta_3)^2 \\ &= -1 \cdot 1 + 0^2 = 1, \qquad \text{as required.} \end{aligned}$$

The remaining properties $JI = -IJ = K$, $KI = -KJ = J$, and $JK = -KJ = I$, all follow from the relations $I^2 = J^2 = K^2 = -1$, $IJ = K$, and the fact that multiplication is associative in $\mathbb{H}$. ■

By inspection, we note that if α_0, α_1, α_2, $\alpha_3 \in \mathbb{R}$ and I, J, and K are as above, and

$$(\alpha_0 1 + \alpha_1 I + \alpha_2 J + \alpha_3 K)I = -I(\alpha_0 1 + \alpha_1 I + \alpha_2 J + \alpha_3 K),$$

then $\alpha_0 = -\alpha_0$ and $\alpha_1 = -\alpha_1$, so $\alpha_0 = \alpha_1 = 0$, while α_2 and α_3 can be arbitrary real numbers. It follows, then, that

$$(\alpha_0 1 + \alpha_1 I + \alpha_2 J + \alpha_3 K)^2 = (\alpha_2 J + \alpha_3 K)^2 = -(\alpha_2^2 + \alpha_3^2)1,$$

so $(\alpha_0 1 + \alpha_1 I + \alpha_2 J + \alpha_3 K)^2$ is a negative real number, or is 0 if and only if $\alpha_0 = \alpha_1 = \alpha_2 = \alpha_3 = 0$. Thus, if we have found elements 1 and I in $\mathbb{H}$ as above, then an element "J" can be found by choosing any nonzero element x of $\mathbb{H}$ such that $xI = -Ix$; then x^2 will be a negative real number, and we can set

$$J = \frac{1}{-\sqrt{-x^2}} \cdot x.$$

As we showed in the proof of the above theorem, K can be set to be IJ; the elements 1, I, J, and K will then have the desired properties.

Since there exists an element I of $M_{2\times2}(\mathbb{R})$, such that $I^2 + 1 = 0$, namely,

$$I = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix},$$

finding such an element I in $\hat{A}$ does not eliminate the possibility that $\hat{A} \cong M_{2\times2}(\mathbb{R})$. However, finding a corresponding element J does guarantee that $A \cong \mathbb{H}$.

Lemma 2.5.16. Let $I \in M_{2\times2}(\mathbb{R})$ such that $I^2 = -1$; then there exists a nonsingular matrix $X \in M_{2\times2}(\mathbb{R})$ such that

$$I = X \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} X^{-1}.$$

Proof. Let

$$I = \begin{bmatrix} \alpha & \beta \\ \gamma & \delta \end{bmatrix} \qquad \text{for } \alpha,\ \beta,\ \gamma,\ \delta \in \mathbb{R}.$$

Then

$$I^2 = \begin{bmatrix} \alpha & \beta \\ \gamma & \delta \end{bmatrix}^2 = \begin{bmatrix} \alpha^2 + \beta\gamma & \beta(\alpha+\delta) \\ \gamma(\alpha+\delta) & \delta^2 + \beta\gamma \end{bmatrix} = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix}.$$

If $\beta = 0$ then $\alpha^2 = \alpha^2 + \beta\gamma = -1$, contradicting the fact that $\alpha \in \mathbb{R}$. Thus $\beta \neq 0$, $\alpha = \delta$, and $\gamma = -(\alpha^2+1)\beta^{-1}$. Thus

$$I = \begin{bmatrix} \alpha & \beta \\ -(\alpha^2+1)\beta^{-1} & -\alpha \end{bmatrix}, \qquad \text{and} \qquad \beta \neq 0.$$

Now if $\beta > 0$, we can set

$$X = \begin{bmatrix} \sqrt{\frac{\beta}{1+\alpha^2}} & -\alpha\sqrt{\frac{\beta}{1+\alpha^2}} \\ 0 & \sqrt{\frac{1+\alpha^2}{\beta}} \end{bmatrix};$$

then

$$X^{-1} = \begin{bmatrix} \sqrt{\frac{1+\alpha^2}{\beta}} & \alpha\sqrt{\frac{\beta}{1+\alpha^2}} \\ 0 & \sqrt{\frac{\beta}{1+\alpha^2}} \end{bmatrix},$$

and

$$\begin{aligned} X \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} X^{-1} &= X \begin{bmatrix} 0 & \sqrt{\frac{\beta}{1+\alpha^2}} \\ -\sqrt{\frac{1+\alpha^2}{\beta}} & -\alpha\sqrt{\frac{\beta}{1+\alpha^2}} \end{bmatrix} \\ &= \begin{bmatrix} \alpha & \beta \\ -\frac{(1+\alpha^2)}{\beta} & -\alpha \end{bmatrix} = I. \end{aligned}$$

Otherwise $\beta < 0$; set

$$X = \begin{bmatrix} \sqrt{\frac{-\beta}{1+\alpha^2}} & -\alpha\sqrt{\frac{-\beta}{1+\alpha^2}} \\ 0 & -\sqrt{\frac{(1+\alpha^2)}{-\beta}} \end{bmatrix};$$

then

$$X^{-1} = \begin{bmatrix} \sqrt{\frac{1+\alpha^2}{-\beta}} & -\alpha\sqrt{\frac{-\beta}{1+\alpha^2}} \\ 0 & -\sqrt{\frac{-\beta}{1+\alpha^2}} \end{bmatrix},$$

and again it is easily checked that

$$X \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} X^{-1} = I, \qquad \text{as desired.} \quad \blacksquare$$

Lemma 2.5.17. Suppose $\hat{A} \cong M_{2\times 2}(\mathbb{R})$.

(i) There do not exist elements I, $J \in \hat{A}$ such that $I^2 = J^2 = -1$ and $IJ = -JI$.

(ii) If $J \in \hat{A}$ such that $J \neq 0$ and $IJ = -JI$, then J has minimal polynomial $t^2 - \zeta^2 = (t+\zeta)(t-\zeta)$, for some nonzero $\zeta \in \mathbb{R}$.

Proof. Suppose 1, $I \in \hat{A}$ with $I^2 + 1 = 0$ and with 1 the multiplicative identity of $\hat{A}$. We first note that there exists an isomorphism $\phi : \hat{A} \to M_{2\times 2}(\mathbb{R})$ with

$$\phi(1) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \qquad \phi(I) = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}.$$

To see this, note that some isomorphism $\psi : \hat{A} \to M_{2\times 2}(\mathbb{R})$ exists; now $\psi(1) = \phi(1)$ as given above, since ψ is an isomorphism of algebras over $\mathbb{R}$. Since $(\psi(I)^2 + 1) = 0$, we conclude (by the previous lemma) that there exists a nonsingular matrix $X \in M_{2\times 2}(\mathbb{R})$ such that

$$\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} = X\psi(I)X^{-1};$$

the isomorphism $\phi : \hat{A} \to M_{2\times 2}(\mathbb{R})$ such that $\phi(a) = X^{-1}\psi(a)X$, for all $a \in \hat{A}$ has the desired properties.

Suppose an element J exists with the properties stated in part (i). Let α, β, γ, $\delta \in \mathbb{R}$ such that

$$\phi(J) = \begin{bmatrix} \alpha & \beta \\ \gamma & \delta \end{bmatrix}.$$

Now since $\phi(I)\phi(J) = -\phi(J)\phi(I)$,

$$\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}\begin{bmatrix} \alpha & \beta \\ \gamma & \delta \end{bmatrix} = -\begin{bmatrix} \alpha & \beta \\ \gamma & \delta \end{bmatrix}\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix},$$

so

$$\begin{bmatrix} \gamma & \delta \\ -\alpha & -\beta \end{bmatrix} = \begin{bmatrix} \beta & -\alpha \\ \delta & -\gamma \end{bmatrix}.$$

Hence $\alpha = -\delta$, $\beta = \gamma$, and

$$\begin{aligned} \phi(J^2) &= (\phi(J))^2 \\ &= \begin{bmatrix} \alpha & \beta \\ \beta & -\alpha \end{bmatrix}^2 \\ &= \begin{bmatrix} \alpha^2 + \beta^2 & 0 \\ 0 & \alpha^2 + \beta^2 \end{bmatrix} \\ &\neq \phi(-1), \end{aligned}$$

since $\alpha^2 + \beta^2 \geq 0$. Further, $\alpha^2 + \beta^2 = 0$ only if $J = 0$. Thus no element J exists with all the properties stated in (i); if $J \neq 0$ and $IJ = -JI$ then we have seen that the minimal polynomial of J divides $t^2 - \zeta^2 = (t + \zeta)(t - \zeta)$ for $\zeta \in \mathbb{R}$ such that $\zeta^2 = \alpha^2 + \beta^2$ (for α, β as above). Clearly J is not a scalar multiple of 1, so the minimal polynomial of J is not a proper divisor of $t^2 - \zeta^2$, proving (ii). ∎

We now have a test which can be used to distinguish between $\mathbb{H}$ and $M_{2\times 2}(\mathbb{R})$. Given an algebra $\hat{A}$ which is isomorphic to one of these two algebras over $\mathbb{R}$, we compute the identity $1 \in \hat{A}$ and an element I of $\hat{A}$ such that $I^2 + 1 = 0$. We then examine a nonzero element $\hat{J}$ of $\hat{A}$ such that $I\hat{J} = -\hat{J}I$, and compute $\hat{J}^2$. If $\hat{A} \cong \mathbb{H}$ then (by Lemma 2.5.15) $\hat{J}$ can be used with I to generate a basis with the usual multiplication table for $\mathbb{H}$. Otherwise, $\hat{A} \cong M_{2\times 2}(\mathbb{R})$, and (by Lemma 2.5.17), I and $\hat{J}$ can be used to compute a pair of nonzero idempotents $E_{1\,1}$ and $E_{2\,2}$ (with $E_{1\,1} + E_{2\,2} = 1$ and $E_{1\,1}E_{2\,2} = E_{2\,2}E_{1\,1} = 0$) in $\hat{A}$. We implement this test in the algorithm on the following pages.

We now have the methods necessary to decompose simple algebras over $\mathbb{R}$. Given a basis for a simple algebra $A \subseteq M_{m\times m}(F)$ for a number field $F \subseteq \mathbb{R}$, with $A \otimes_F \mathbb{R}$ simple over $\mathbb{R}$, we use a probabilistic Boolean algorithm to compute integers k, $l > 0$ such that $A \otimes_F \mathbb{R} \cong M_{k\times k}(D)$, for a division algebra D of dimension l over $\mathbb{R}$ (so $l = 1$, 2, or 4). We also compute elements of a standard basis $e_{i\,j}$ and elements $d_1 e_{i\,j}$, $d_2 e_{i\,j}$, …, $d_l e_{i\,j}$, where d_1, d_2, …, $d_l \in A \otimes_F \mathbb{R}$ form a basis for a subalgebra of $A \otimes_F \mathbb{R}$ isomorphic to the division algebra D, with $d_r e_{i\,j} = e_{i\,j} d_r$, for $1 \leq r \leq l$ and $1 \leq i$, $j \leq k$. The elements $e_{i\,j}$ and $d_r e_{i\,j}$ will all lie in $A \otimes_F E_{i\,j}$, for $E_{i\,j} \subseteq \mathbb{R}$ an algebraic extension with dimension $k^{O(1)}$ over F.

Unfortunately, we cannot guarantee that the elements

$$d_r = d_r e_{1\,1} + d_r e_{2\,2} + \cdots + d_r e_{k\,k}$$

(for $1 \leq r \leq l$) lie in $A \otimes_F E_r$ for a number field E_r of small dimension over F. While numerical estimates of the entries of the matrices d_r can be computed (to arbitrarily high precision) using the output generated by our algorithm, this does not establish that "symbolic" representations (including the minimal polynomials over $\mathbb{Q}$ for these algebraic numbers) can be computed (for any standard basis $e_{i\,j}$, $1 \leq i$, $j \leq k$) in polynomial time.

For the sake of brevity (and readability), we merely sketch our algorithm. Input and output are stated in detail; we claim (without further proof) that the output can be computed using the methods sketched for "Decomposition of a Simple Algebra over $\mathbb{R}$".

Algorithm **Decomposition of Quaternion Algebras over $\mathbb{R}$**

Input.

- Integers h, $m > 0$.
- The coefficients of a monic irreducible polynomial $f \in \mathbb{Z}[t]$
  $f = t^h + \lambda_{h-1}t^{h-1} + \cdots + \lambda_1 t + \lambda_1$ of degree h.
- Endpoints of an isolating interval in $\mathbb{R}$ for a real root α of f.
- Matrices a_1, a_2, a_3, $a_4 \in M_{m\times m}(F)$, for $F = \mathbb{Q}[\alpha] \subseteq \mathbb{R}$, which form the basis of a finite-dimensional simple associative algebra $A \subseteq M_{m\times m}(F)$ of dimension 4 over F, whose centre has dimension 1 over F, such that $A \otimes_F \mathbb{R}$ is simple over $\mathbb{R}$.

Output.

- Integer $\hat{h} > 0$, the coefficients of a monic irreducible polynomial
  $\hat{f} = t^{\hat{h}} + \hat{\lambda}_{\hat{h}-1}t^{\hat{h}-1} + \cdots + \hat{\lambda}_1 + \hat{\lambda}_0 \in \mathbb{Z}[t]$ of degree $\hat{h}$, and
  endpoints of an isolating interval in $\mathbb{R}$ for a real root $\hat{\alpha}$ of $\hat{f}$.
- A polynomial $g \in \mathbb{Q}[t]$ with degree less than $\hat{h}$ such that $g(\hat{\alpha}) = \alpha$.
- Integer $l > 0$, with value 1 if $A \otimes_F \mathbb{R} \cong M_{2\times 2}(\mathbb{R})$, and with value 4 if $A \otimes_F \mathbb{R} \cong \mathbb{H}$.
  — If $l = 1$:
  Matrices $E_{1\,1}$, $E_{1\,2}$, $E_{2\,1}$, $E_{2\,2} \in A \otimes_F \hat{F}$, forming a basis for $A \otimes_F \hat{F}$, such that $E_{1\,1} + E_{2\,2} = 1$, and $E_{r\,s}E_{t\,u} = \delta_{s\,t}E_{r\,u}$, for $1 \le r$, s, t, $u \le 2$.
  — If $l = 4$:
  Matrices 1, I, J, $K \in A \otimes_F \hat{F}$, forming a basis for $A \otimes_F \hat{F}$, such that $\alpha 1 = 1\alpha = \alpha$ for all $\alpha \in A \otimes_F \hat{F}$,
  $I^2 = J^2 = K^2 = -1$, $IJ = -JI = K$, $JK = -KJ = I$,
  and $KI = -IK = J$.

(1) Solving a system of linear equations, compute the identity 1 (as a linear combination of a_1, a_2, a_3, a_4) in A. Set $\iota \in A$ to be any element which is not a scalar multiple of 1.

(2) Compute the minimal polynomial, $t^2 + \gamma_1 t + \gamma_0$, of ι over F.

(3) If $\gamma_1^2 - 4\gamma_0 \ge 0$ then the minimal polynomial of ι is reducible, and $A \otimes_F \mathbb{R} \cong M_{2\times 2}(\mathbb{R})$. Set $\kappa = \iota + (\gamma_1/2)$, so κ has minimal polynomial $t^2 + \bar{\gamma} = t^2 - (\gamma_1^2 - 4\gamma_0)/4$ (and $\bar{\gamma} < 0$). Set $\beta = \alpha$, $E = F$, and go to step 6.

(4) Use the algorithms of Loos [87] to compute the minimal polynomial and an isolating interval for an algebraic number $\beta \in \mathbb{R}$ such that $F \subseteq E = \mathbb{Q}[\beta] = \mathbb{Q}[\alpha,\, \hat{\gamma}]$, where $\hat{\gamma}$ is a real algebraic number such that $\hat{\gamma}^2 = 4\gamma_0 - \gamma_1^2 \in F$, and to compute a polynomial $\hat{g} \in \mathbb{Q}[t]$ with small degree such that $\hat{g}(\beta) = \alpha$.
Set $I = \hat{\gamma}^{-1}(2\iota + \gamma_1) \in A \otimes_F E$; then $I^2 + 1 = 0$.

(5) Solve a system of linear equations over E to compute a nonzero element κ of $A \otimes_F E$ such that $I\kappa = -\kappa I$. Compute the minimal polynomial, $t^2 + \bar{\gamma}$, of κ (see Lemmas 2.5.15–2.5.17).
If $\bar{\gamma} < 0$, then $A \otimes_F \mathbb{R} \cong M_{2\times 2}(\mathbb{R})$; go to step 6.
Otherwise, $\bar{\gamma} > 0$ and $A \otimes_F \mathbb{R} \cong \mathbb{H}$; go to step 9.

Case 1: $A \otimes_F \mathbb{R} \cong M_{2\times 2}(\mathbb{R})$.

(6) Set $l = 1$. Use the algorithms of Loos [87] to compute the minimal polynomial $\hat{f}$ (with degree $\hat{h}$ over $\mathbb{Q}$), and an isolating interval in $\mathbb{R}$, for an algebraic number $\hat{\alpha} \in \mathbb{R}$ such that $F \subseteq E \subseteq \hat{F} = \mathbb{Q}[\hat{\alpha}] = \mathbb{Q}[\beta,\, \tilde{\gamma}]$, where $\tilde{\gamma}$ is a real algebraic number such that $\tilde{\gamma}^2 = -\bar{\gamma} \neq 0$, and to compute a polynomial $g \in \mathbb{Q}[t]$ with degree less than $\hat{h}$ such that $g(\hat{\alpha}) = \alpha$. Then κ has minimal polynomial $(t + \tilde{\gamma})(t - \tilde{\gamma})$ in $\hat{F}[t]$.

(7) Set $E_{1\,1} = (\kappa + \tilde{\gamma})/(2\tilde{\gamma})$ and set $E_{2\,2} = (\kappa - \tilde{\gamma})/(-2\tilde{\gamma})$.
Then $E_{1\,1}^2 = E_{1\,1}$, $E_{2\,2}^2 = E_{2\,2}$, $E_{1\,1} + E_{2\,2} = 1$, and $E_{1\,1}E_{2\,2} = E_{2\,2}E_{1\,1} = 0$.

(8) Compute as $E_{1\,2}$ any nonzero element x of $A \otimes_F \hat{F}$ such that $E_{1\,1}x = xE_{2\,2} = x$ and $xE_{1\,1} = E_{2\,2}x = 0$.
Compute as $E_{2\,1}$ the unique element y of $A \otimes_F \hat{F}$ such that $E_{1\,1}y = yE_{2\,2} = 0$, $yE_{1\,1} = E_{2\,2}y = y$, $yE_{1,2} = E_{2\,2}$, and $E_{1\,2}y = E_{1\,1}$. Return these values, and stop.

Case 2: $A \otimes_F \mathbb{R} \cong \mathbb{H}$.

(9) Set $l = 4$. Use the algorithms of Loos [87] to compute the minimal polynomial $\hat{f}$ (with degree $\hat{h}$) over $\mathbb{Q}$, and an isolating interval in $\mathbb{R}$, for an algebraic number $\hat{\alpha} \in \mathbb{R}$ such that $F \subseteq E \subseteq \hat{F} = \mathbb{Q}[\hat{\alpha}] = \mathbb{Q}[\beta,\, \tilde{\gamma}]$ where $\tilde{\gamma}$ is a real algebraic number such that $\tilde{\gamma}^2 = \bar{\gamma} > 0$, and to compute a polynomial $g \in \mathbb{Q}[t]$ with degree less than $\hat{h}$ such that $g(\hat{\alpha}) = \alpha$.

(10) Now κ has minimal polynomial $t^2 + \tilde{\gamma}^2$ in $\hat{F}[t]$. Set $J = \kappa/\tilde{\gamma}$; then $IJ = -JI$ and $J^2 + 1 = 0$.

(11) Set $K = IJ \in A \otimes_F \hat{F}$. Return the desired values, and stop.

Algorithm **Decomposition of a Simple Algebra over $\mathbb{R}$**

Input.
- Integers h, n, $m > 0$.
- The coefficients of a monic irreducible polynomial $f \in \mathbb{Z}[t]$
  $f = t^h + \lambda_{h-1}t^{h-1} + \cdots + \lambda_1 t + \lambda_0$ of degree h.
- Endpoints of an isolating interval in $\mathbb{R}$ for a real root α of f.
- Matrices $a_1, a_2, \ldots, a_n \in M_{m\times m}(F)$, for $F = \mathbb{Q}[\alpha] \subseteq \mathbb{R}$, which form the basis for a finite-dimensional simple associative algebra $A \subseteq M_{m\times m}(F)$ of dimension n over F, such that $A \otimes_F \mathbb{R}$ is simple over $\mathbb{R}$.
- Error tolerance $\epsilon > 0$.

Output. EITHER:
- Integers k, $l > 0$, such that $A \otimes_F \mathbb{R} \cong M_{k\times k}(D)$, for a division algebra $D \subseteq A \otimes_F \mathbb{R}$ of dimension l over $\mathbb{R}$.
- For integers i and j, such that $1 \le i, j \le k$:

— Integer $h_{i\,j} > 0$, the coefficients of a monic irreducible polynomial
$$f_{i\,j} = t^{h_{i\,j}} + \lambda_{i,\,j,\,h_{i\,j}-1}t^{h_{i\,j}-1} + \cdots + \lambda_{i,\,j,\,1}t + \lambda_{i,\,j,\,0} \in \mathbb{Z}[t]$$
of degree $h_{i\,j}$, and the endpoints of an isolating interval in $\mathbb{R}$, for a real root $\alpha_{i\,j}$ of $f_{i\,j}$, such that $F \subseteq E_{i\,j} \subseteq \mathbb{R}$, for $E_{i\,j} = \mathbb{Q}[\alpha_{i\,j}]$.

— Coefficients of a polynomial
$$g_{i\,j} = \zeta_{i,\,j,\,h_{i\,j}-1}t^{h_{i\,j}-1} + \cdots + \zeta_{i,\,j,\,1}t + \zeta_{i,\,j,\,0} \in \mathbb{Q}[t]$$
such that $\alpha = g_{i\,j}(\alpha_{i\,j})$.

— Elements $\nu_{i\,j\,s}$ and $\mu_{i\,j\,r\,s}$ of $E_{i\,j}$, for $1 \le r \le l$ and $1 \le s \le n$, defining elements $e_{i\,j} = \sum_{s=1}^n \nu_{i\,j\,s}a_s$ and $d_{r\,i\,j} = \sum_{s=1}^n \mu_{i\,j\,r\,s}a_s$ of $A \otimes_F E_{i\,j}$ so that, for $d_r = \sum_{i=1}^k d_{r\,i\,i} \in A \otimes_F \mathbb{R}$, $d_{r\,i\,j} = d_r e_{i\,j}$, and these elements have the following properties.

(1) $d_1, d_2, \ldots, d_l$ is a basis for the division algebra $D \subseteq A \otimes_F \mathbb{R}$ over $\mathbb{R}$;

(2) the elements $e_{i\,j}$ form a basis for $A \otimes_F \mathbb{R} \cong M_{k\times k}(D)$ over D, with $e_{1\,1} + e_{2\,2} + \cdots + e_{k\,k} = 1$ and $e_{r\,s}e_{t\,u} = \delta_{s\,t}e_{r\,u}$ for $1 \le r, s, t, u \le k$;

(3) $d_r e_{i\,j} = e_{i\,j}d_r$ for $1 \le r \le l$ and $1 \le i, j \le k$; and

(4) the elements $d_{r\,i\,j} = d_r e_{i\,j}$ (for $1 \le r \le l$, $1 \le i, j \le k$) form form a basis for $A \otimes_F \mathbb{R}$ over $\mathbb{R}$.

OR: *failure*, with probability less than ϵ.

(1) If $n = 2k^2$ for some $k \in \mathbb{Z}$ then $D \cong \mathbb{C}$; go to step 2.
Otherwise, $D \cong \mathbb{R}$ or $D \cong \mathbb{H}$; go to step 6.

Case 1. Centre$(A \otimes_F \mathbb{R}) \cong \mathbb{C}$; $D \cong \mathbb{C}$.

(2) Compute a basis for Centre(A) over F; this will have dimension 2 over F. Set ι to be an element of Centre(A) which is not a scalar multiple of 1. Use ι to compute an element I of Centre$(A \otimes_F E)$, for E an extension of dimension at most 2 over F, such that $I^2 + 1 = 0$. (See steps 2–4 of algorithm "Decomposition of Quaternion Algebras over $\mathbb{R}$").

(3) Compute a basis $\hat{a}_1, \hat{a}_2, \ldots, \hat{a}_{k^2}$ (with $n/2 = k^2$ elements) for $A \otimes_F E[I]$ over the field $E[I]$ (isomorphic to $E[\sqrt{-1}]$, a number field of dimension 2 over E, with $I = \sqrt{-1}$ a square root of -1); $\hat{a}_1, I\hat{a}_1, \hat{a}_2, I\hat{a}_2, \ldots, \hat{a}_{k^2}, I\hat{a}_{k^2}$ is then a basis for $A \otimes_F E$ over E. Compute a set of structure constants (in $E[\sqrt{-1}]$, replacing I by $\sqrt{-1}$) for $A \otimes_F E[\sqrt{-1}]$ with respect to the basis $\hat{a}_1, \hat{a}_2, \ldots, \hat{a}_{k^2}$.

(4) Use the regular matrix representation for $A \otimes_F E[\sqrt{-1}]$ over $E[\sqrt{-1}]$, with respect to the basis computed in step 3, as input for the algorithm "Decomposition of a Simple Algebra over $\mathbb{C}$", to compute a standard basis for $(A \otimes_F E[\sqrt{-1}]) \otimes_{E[\sqrt{-1}]} \mathbb{C} \cong A \otimes_F \mathbb{C}$ over $\mathbb{C}$.

(5) Use the standard basis (and number fields) computed for $A \otimes_F \mathbb{C}$ in step 4 to recover the desired basis for $A \otimes_F \mathbb{R}$:

- For each number field $G = \mathbb{Q}[\beta]$ returned in step 4, compute a generator $\hat{\beta}$ for the number field $\hat{G} = \mathbb{Q}[\hat{\beta}] = \mathbb{Q}[\mathrm{Re}(\beta), \mathrm{Im}(\beta)] \subseteq \mathbb{R}$.
- For each element λ of G included in the output, compute the elements $\mathrm{Re}(\lambda)$ and $\mathrm{Im}(\lambda)$ of $\hat{G}$. (Now each element $\sum_{j=1}^{k^2} \lambda_j \hat{a}_j$ of $(A \otimes_F E[\sqrt{-1}]) \otimes_{e[\sqrt{-1}]} G \cong A \otimes_F G$ can be replaced by the element $\sum_{j=1}^{k^2} \mathrm{Re}(\lambda_j)\hat{a}_j + \sum_{j=1}^{k^2} \mathrm{Im}(\lambda_j) I\hat{a}_j$ of $A \otimes_F \hat{G} \subseteq A \otimes_F \mathbb{R}$.)

This is sufficient to produce a basis $d_1 = 1$, $d_2 = I \in A \otimes_F E$ for $D \subseteq A \otimes_F \mathbb{R}$, as well as the elements of a standard basis for $A \otimes_F \mathbb{R}$ over D, as required. Return the desired values, and stop.

Case 2. Centre$(A \otimes_F \mathbb{R}) \cong \mathbb{R}$; *$D \cong \mathbb{R}$ or $D \cong \mathbb{H}$.*

(6) Set $\hat{k} = \sqrt{n} \in \mathbb{Z}$. Choose elements $\lambda_1, \lambda_2, \ldots, \lambda_n$ randomly and independently from a subset I of F, of size $\lceil \hat{k}(2\hat{k} - 1)\epsilon^{-1} \rceil$.

(7) Compute the (monic) minimal polynomial $\hat{f}$ in $F[t]$ of the element
$a = \lambda_1 a_1 + \lambda_2 a_2 + \cdots + \lambda_n a_n$ of A.
If this polynomial has degree less than $\hat{k}$, or has degree $\hat{k}$ and is not squarefree, report *failure*, and stop. Otherwise, continue.

(8) Use factorisation (of $\hat{f}$) over $\mathbb{R}$ to compute a set of number fields $E_1, E_2, \ldots, E_t \subseteq \mathbb{R}$, each a (small) algebraic extension of F, such that each polynomial ring $E_i[t]$ contains an irreducible factor of $\hat{f}$ over $\mathbb{R}$.

(9) Use the factorisation of $\hat{f}$ computed in step 8 to generate a set of idempotents $e_1, e_2, \ldots, e_t \in A \otimes_F \mathbb{R}$ (with $e_i \in A \otimes_F E_i$ for $1 \le i \le t$; *cf.* algorithm "Extraction of Idempotents and steps 4–5 of "Extraction of Simple Components over an Extension" of Section 2.4).
Now for each i, either $e_i(A \otimes_F \mathbb{R})e_i \cong \mathbb{R}$, $e_i(A \otimes_F \mathbb{R})e_i \cong M_{2\times 2}(\mathbb{R})$, or $e_i(A \otimes_F \mathbb{R})e_i \cong \mathbb{H}$.

(10) For each i, with $1 \le i \le t$, such that $e_i(A \otimes_F E_i)e_i$ has dimension 4 over E_i, use the algorithm "Decomposition of Quaternion Algebras over $\mathbb{R}$" to decide whether $e_i(A \otimes_F \mathbb{R})e_i \cong \mathbb{H}$ or $e_i(A \otimes_F \mathbb{R})e_i \cong M_{2\times 2}(\mathbb{R})$, and to compute an appropriate basis for $e_i(A \otimes_F \mathbb{R})e_i$ (with entries in a small extension of E_i).

(11) The values computed in steps 10 and 11 include the values of the integers k and l, a complete set of idempotents $e_{1\,1}, e_{2\,2}, \ldots, e_{k\,k}$ in $A \otimes_F \mathbb{R}$, and the elements $d_r e_{i\,i}$ of $A \otimes_F \mathbb{R}$, for $1 \le r \le l$ (and $d_1, d_2, \ldots, d_l$ a basis for D over $\mathbb{R}$) and $1 \le i \le k$. Solve systems of equations (over small extensions of F) to compute the remaining elements $e_{i\,j}$ of a standard basis for $A \otimes_F \mathbb{R}$ over D, and for the elements $d_{r\,i\,j} = d_r e_{i\,j} = (d_r e_{i\,i}) \cdot e_{i\,j}$ for $1 \le i, j \le k$ and $1 \le r \le l$.
Return the desired values, and stop.

Theorem 2.5.18. Suppose $A \subseteq M_{m\times m}(F)$ is a simple associative algebra over a number field $F = \mathbb{Q}[\alpha] \subseteq \mathbb{R}$, such that $A \otimes_F \mathbb{R}$ is simple over $\mathbb{R}$, with $A \cong M_{k\times k}(D)$ for $k > 0$ and for a division algebra D over $\mathbb{R}$. Then there exist bases $d_1 = 1, d_2, \ldots, d_l$ for $D \subseteq A \otimes_F \mathbb{R}$ over $\mathbb{R}$, and $e_{i\,j}$ (for $1 \leq i, j \leq k$) for $A \otimes_F \mathbb{R}$ over D, such that $e_{i\,j} = d_1e_{i\,j}, d_2e_{i\,j}, \ldots, d_le_{i\,j} \in A \otimes_F E_{i\,j}$, with $F \subseteq E_{i\,j} = \mathbb{Q}[\alpha_{i\,j}] \subseteq \mathbb{R}$ and $E_{i\,j}$ a finite algebraic extension, with dimension at most $4k$ over F. Given the minimal polynomial over $\mathbb{Q}$ and an isolating interval in $\mathbb{R}$ for α, a basis $a_1, a_2, \ldots, a_n$ for A over F, and an integer $N > 0$ (in binary notation) as input, a probabilistic Boolean algorithm can be used to compute a generator $\alpha_{i\,j}$ of a field $E_{i\,j}$ as above, as well as the elements $d_1e_{i\,j}, d_2e_{i\,j}, \ldots, d_le_{i\,j}$ of $A \otimes_F E_{i\,j}$, using time polynomial in the input size, and with probability of failure less than $\epsilon = N^{-1}$.

Proof. We first note that the correctness of the algorithm "Decomposition of Quaternion Algebras over $\mathbb{R}$", for the decomposition of simple algebras of dimension 4 with centre isomorphic to $\mathbb{R}$, is implied by Lemmas 2.5.15–2.5.17. It is clear that this algorithm can be implemented as a Boolean algorithm which uses time polynomial in the size of its input.

Now correctness of the algorithm "Decomposition of a Simple Algebra over $\mathbb{R}$" for algebras with centre of dimension one over $\mathbb{R}$, and the bound on the probability of failure, follow from Lemma 2.5.14 and the result of Schwartz (Proposition 2.4.23). Correctness for the only other algebras which can occur as input, those with centre of dimension two over $\mathbb{R}$ (isomorphic to $\mathbb{C}$), follows from the correctness of the algorithm "Decomposition of a Simple Algebra over $\mathbb{C}$" (see Theorem 2.5.11).

Timing analysis for the algorithms, and the proof of the upper bound stated in the theorem for the dimension of the number fields $E_{i\,j}$ over F, are straightforward. ∎

Again, the methods of Babai and Rónyai [5] can be used to improve the results stated here. Instead of computing a set of field extensions $\{E_{i\,j}\}$ and using all of these to represent a standard basis, we can use the probabilistic methods already discussed to obtain a left ideal in $A \otimes_F E$, for $E \subseteq \mathbb{R}$ such that the degree of E over F is small, which is either an irreducible left ideal or the direct sum of two irreducible left ideals. In the latter case the techniques used in the algorithm "Decomposition of Quaternion Algebras over $\mathbb{R}$" can be employed to extract a single irreducible left ideal over $A \otimes_F \hat{E}$, where $\hat{E}$ is an extension with degree at most two over E (and, again, $\hat{E} \subseteq \mathbb{R}$). The methods discussed in [5] (and, here, at the end of Section 2.5.3) can then be used with this ideal to obtain a complete basis for $A \otimes_F E$ over E (or for $A \otimes_F \hat{E}$ over $\hat{E}$), of the desired form, in probabilistic polynomial time.

2.5.5. Simple Algebras over Number Fields

Finally, we consider the problem of decomposing simple algebras over algebraic number fields. Unlike the problems discussed in previous sections, no efficient deterministic or probabilistic algorithm is known for this problem. Ronyai [103] considers the simplest nontrivial case — simple algebras of dimension 4 over $\mathbb{Q}$ — and provides strong evidence that no efficient (i.e., polynomial time) algorithm exists for decomposition, even for this simple case. We (briefly) consider the question of whether deterministic algorithms exist for the more general problem.

We consider several "decision problems" (problems with yes/no answers) related to the decomposition of simple algebras.

Problem 1. Given a simple algebra A over a finite algebraic extension F of $\mathbb{Q}$, decide whether A is a division algebra.

Problem 2. Given a simple algebra A over a finite algebraic extension F of $\mathbb{Q}$, decide whether $A \cong M_{k\times k}(F)$, for some integer $k > 0$.

Problem 3. Given two simple algebras A_1 and A_2 over a finite algebraic extension F of $\mathbb{Q}$, decide whether $A_1 \cong A_2$.

We first note (as does Rónyai) that if we are considering algorithms which are correct over arbitrary number fields F, with a description of $F = \mathbb{Q}[\alpha]$ as part of the input for the problem (rather than algorithms correct only for a specific, fixed, number field F), then we can assume without loss of generality that $\text{Centre}(A) \cong F$. For if A is a simple algebra over F whose centre has dimension greater than one over F, then $\text{Centre}(A) \cong E$, a finite algebraic extension field of F. Computing a basis and set of structure constants for A over $\text{Centre}(A) \cong E$, we obtain a description of a simple algebra $\bar{A}$ over E whose centre is isomorphic to E. If $A \cong M_{k\times k}(D)$ for $k > 0$ and for a division algebra D over F, then $\text{Centre}(D) \cong E$, and $\bar{A} \cong M_{k\times k}(\bar{D})$ for the same integer $k > 0$, and for a division algebra $\bar{D}$ over E. Further, if $\bar{d}_1, \bar{d}_2, \ldots, \bar{d}_h$ is a basis for $\bar{D}$ over E, and $c_1, c_2, \ldots, c_l$ is a basis for E over F, then D is isomorphic to a division algebra over F with basis $\{\, \bar{d}_i c_j \;:\; 1 \le i \le h \text{ and } 1 \le j \le l \,\}$, of dimension hl over F. That is, $D \cong E \otimes_F \bar{D}$, regarded as an algebra over F. We decompose A over F by computing a description of the simple algebra $\bar{A}$ over E, decomposing $\bar{A}$, and then using the idempotents in $\bar{A}$ and the basis for $\bar{D}$ generated by this process to recover a set of primitive idempotents for A over F and a basis for D over F. (This method was used in Section 2.4, to decompose simple algebras over $\mathbb{R}$ whose centres were isomorphic to $\mathbb{C}$.) Henceforth, we assume that the simple algebra A is *central* over F: that is, the centre has dimension one over, and is isomorphic to, the ground field F.

We first note that a division algebra A over F cannot be distinguished from a matrix ring $M_{k\times k}(F)$ over F by comparison of dimension over F.

Proposition 2.5.19. Let A be a finite-dimensional central simple algebra over a number field F; then A has dimension k^2 over F, for some integer $k > 0$.

If $k = 1$, then $A \cong F$, and A is a division algebra over F. Hence the first case for which the problem of deciding whether a central simple algebra A is a division algebra is nontrivial, is the case $k = 2$ and the dimension of A over F is $n = k^2 = 4$ (for there exist division algebras of dimension 4 over $\mathbb{Q}$, as well as the simple algebra $M_{2\times 2}(\mathbb{R})$, which is clearly not a division algebra). Rónyai [103] gives evidence that this simplest case is difficult, as explained below.

Definition 2.5.20. Let $n > 1$ be a positive integer, and let $m > 0$ be an integer such that $m < n$ and $\gcd(m, n) = 1$. The integer m is a *quadratic residue modulo* n if there exists an integer $r \in \mathbb{Z}$ such that $0 < r < n$ and $r^2 \equiv m \bmod n$; otherwise, m is a *quadratic nonresidue modulo* n. The *quadratic residuosity problem* is the problem of deciding, given binary representations of integers m and n, whether m is a quadratic residue modulo n.

Related to this decision problem is the computational problem of computing an integer r such that $r^2 \equiv m \bmod n$, given binary representations of an integer $n > 0$ and a quadratic residue m modulo n. There is a *Las Vegas* algorithm (that is, a probabilistic algorithm which either returns a correct answer, or signals *failure*, the latter with arbitrarily small positive probability) which takes as input a positive integer n (in binary) and returns positive integers n_1, $n_2 > 1$ such that $n_1 n_2 = n$, if n is not prime, and which uses time polynomial in its input size ($O(\log n) + O(\log \epsilon^{-1})$, for error tolerance $\epsilon > 0$), plus the time required to compute a square root (modulo n) for a quadratic residue modulo n. Dixon [36] uses this approach to obtain a probabilistic algorithm which splits a composite integer into two nontrivial factors, with arbitrarily high probability of success, using time sublinear in n. Now the problem of factoring an arbitrary positive integer n (given its binary representation) is conjectured to be hard: It is believed that no (deterministic or probabilistic) Boolean algorithm can be used to factor n using time polynomial in its input size $O(\log n)$. A number of cryptographic protocols have been based on this belief (see, for example, Rabin [98], and Goldwasser and Micali [59]). If this belief is correct, then the problem of computing square roots of quadratic residues modulo n for an arbitrary integer n must be hard as well. Rónyai reduces the problem of deciding whether an integer m is a quadratic residue modulo n, for n an arbitrary squarefree integer, to the problem of deciding whether a central simple algebra of dimension 4 over $\mathbb{Q}$ is a division algebra. He also reduces the problem of finding square roots of quadratic residues modulo n (for n squarefree), and hence the problem of factoring squarefree integers, to the problem of computing zero divisors in central simple algebras of dimension 4 over $\mathbb{Q}$.

Proposition 2.5.21 (Rónyai [103], [104]). Assuming the Generalised Riemann Hypothesis (GRH), there exists a Las Vegas polynomial time reduction from the problem of deciding whether a positive integer m is a quadratic residue modulo a squarefree integer n, to deciding whether a central simple division algebra of dimension 4 over $\mathbb{Q}$ is isomorphic to $M_{2\times 2}(\mathbb{Q})$.

Proposition 2.5.22 (Rónyai [103], [104]). Assuming GRH, there exists a randomised polynomial time reduction from the problem of factoring squarefree integers to the problem of finding zero divisors in central simple algebras of dimension 4 over $\mathbb{Q}$.

Propositions 2.5.21 and 2.5.22 are proved in [103] (as Theorem 4.5 and Corollary 4.6). As noted above, the reductions are conditional on the Generalised Riemann Hypothesis; further information about this hypothesis can be found in Bach [6].

These facts provide evidence that no efficient (that is, polynomial time) algorithms for decomposition of simple algebras over number fields exist. Fortunately, there is some good news.

Proposition 2.5.23 (Rónyai [103], [104]). The problem of deciding whether a central simple algebra A of dimension 4 over $\mathbb{Q}$ is isomorphic to $M_{2\times 2}(\mathbb{Q})$ is in $\mathrm{NP} \ \cap \ \mathrm{co}-\mathrm{NP}$.

Among other things, Proposition 2.5.23 implies that we can decide whether a central simple algebra of dimension 4 over $\mathbb{Q}$ is a division algebra, using a deterministic algorithm in polynomial time, provided we are also given some extra information (of polynomial size) — so that there exists a polynomial-size *certificate* (or *proof*) for a division algebra of dimension 4 over $\mathbb{Q}$, as well as for an algebra isomorphic to $M_{2\times 2}(\mathbb{Q})$. It also implies that there is a deterministic algorithm for this decision problem (again, for central simple algebras of dimension 4 over $\mathbb{Q}$) which uses time exponential in the size of the input. For a more detailed discussion of the complexity classes NP and $\mathrm{co}-\mathrm{NP}$, and the implications of membership of a decision problem in $\mathrm{NP} \cap \mathrm{co}-\mathrm{NP}$, see Garey and Johnson [51].

We have seen that there is reason to believe that no efficient (polynomial time) algorithms exist for the computational problem "Decomposition of a Simple Algebra" over a number field $F = \mathbb{Q}[\alpha]$, or for the decision problems stated at the beginning of this section. It seems natural to ask whether any algorithms exist for these problems at all.

Question 2.5.24. Does there exist an algorithm for the decomposition of finite-dimensional simple associative algebras over algebraic number fields? Are the decision problems of deciding whether such an algebra is a division algebra, of deciding whether a simple associative algebra A over a number field F is isomorphic to $M_{k\times k}(F)$ for some $k > 0$, and of deciding whether two such algebras are isomorphic, decidable problems?

We will not answer this question. As a partial solution, we state some relationships between the decision problems given above.

Clearly decision problem 2 is a special case of problem 3 — so that if we have an efficient solution for the latter problem, then we have one for the former, as well. We will show that the converse of this is true as well.

Definition 2.5.25. Let A be a finite-dimensional associative algebra of dimension n over F. The *opposite algebra* of A, A^*, is the algebra having the same structure as A as a vector space over F (so that x, $y \in A^*$ if x, $y \in A$, and addition and multiplication by elements of F are the same in A^* as they are in A), with the product of two elements x and y in A^* defined as

$$x \circ y = yx,$$

where xy denotes the product of x and y in A.

In Section 2.2.3, we (briefly) discussed the *tensor product* of algebras over F. As we note there, if A and B are both associative algebras over a field F then $A \otimes_F B$ is also an associative algebra over F, with multiplication defined "componentwise" (see Section 2.2.3 for details).

We now note some additional properties of tensor products of associative algebras.

Proposition 2.5.26. Suppose A, B, and C are finite-dimensional associative algebras over F, D is a finite-dimensional division algebra over F, A has dimension $n > 0$, and that k, $l > 0$.

(i) $A \otimes_F F \cong A$.

(ii) $A \otimes_F (B \otimes_F C) \cong (A \otimes_F B) \otimes_F C$.

(iii) $A \otimes_F B \cong B \otimes_F A$.

(iv) $M_{k\times k}(F) \otimes_F M_{l\times l}(D) \cong M_{kl\times kl}(D)$.

(v) If A and B are simple then $A \otimes_F B$ is simple.

(vi) If A and B are central simple then $A \otimes_F B$ is central simple.

(vii) If A is central simple then $A \otimes_F A^* \cong M_{n\times n}(F)$;

(viii) If A and B are central simple then $A \otimes_F B^* \cong M_{m\times m}(F)$ for some $m > 0$ if and only if there exists a central division algebra D over F and integers r, $s > 0$ such that $A \cong M_{r\times r}(D)$ and $B \cong M_{s\times s}(D)$.

These facts are well known; for a proof see, for example, Pierce [95] (see Sections 12.4 and 12.5). They establish the fact that the central simple algebras over F are closed with respect to the formation of tensor products. Now we say that finite-dimensional central simple algebras A and B are *equivalent*, and write $A \sim B$, if there exist positive integers r and s such that $A \otimes_F M_{r\times r}(F) \cong B \otimes_F M_{s\times s}(F)$. Propositions 2.5.26 (i)–(iv) prove that "$\sim$" is an equivalence relation. If we write $[A]$ as the class of central simple algebras over F equivalent to A, then we conclude (using Proposition 2.5.26 (vii) and (viii)) that each equivalence class $[A]$ contains a finite-dimensional division algebra $D \sim A$, which is unique up to isomorphism of associative algebras over F. It is also clear that if $A_1 \sim A_2$ and $B_1 \sim B_2$ then $A_1 \otimes_F B_1 \sim A_2 \otimes_F B_2$, for arbitrary central simple associative algebras A_1, A_2, B_1, B_2 over F. Thus $\otimes_F$ defines a binary operation on equivalence classes. By Proposition 2.5.26, this operation has the following properties.

(i) $([A] \otimes_F [B]) \otimes_F [C] = [A] \otimes_F ([B] \otimes_F [C])$.

(ii) $[A] \otimes_F [F] = [F] \otimes_F [A] = [A]$.

(iii) $[A] \otimes_F [A^*] = [A^*] \otimes_F [A] = [F]$.

(iv) $[A] \otimes_F [B] = [B] \otimes_F [A]$.

Thus the set of equivalence classes of central simple algebras over F forms an Abelian group with respect to the operation $\otimes_F$, with identity $[F]$, and such that $[A]^{-1} = [A^*]$. For a further discussion of this group, called the *Brauer group* of the field F, see Pierce [95]. If A and B are arbitrary finite-dimensional central simple algebras, then $A \cong B$ if and only if A and B have the same dimension over F and $[A] = [B]$. The latter condition is equivalent to the condition that $A \otimes_F B^*$ be isomorphic to $M_{r\times r}(F)$ for some $r > 0$ (that is, that $[A]^{-1} = [B]^{-1} = [B^*]$, and hence that $[A] \otimes_F [B^*] = [F]$). Given bases and sets of structure constants for central simple algebras A and B, we obtain a basis and set of structure constants for B^* by using the same basis as for B and reordering the structure constants (specifically, if $\{\, \gamma_{i\,j\,k} \,\}$ are structure constants for B, and $\{\, \hat{\gamma}_{i\,j\,k} \,\}$ are structure constants for B^* with respect to the same basis, then $\hat{\gamma}_{i\,j\,k} = \gamma_{j\,i\,k}$). A basis and set of structure constants for $A \otimes_F B^*$ are then easily obtained. Since we have already noted that it is sufficient to consider central simple algebras when discussing the decision problems stated at the beginning of this section, we have proved the following result.

Theorem 2.5.27. The decision problems of deciding whether a finite-dimensional simple algebra A of dimension $n = k^2$ over a number field $F = \mathbb{Q}[\alpha]$ is isomorphic to $M_{k\times k}(F)$, and of deciding whether two finite-dimensional simple algebras A_1 and A_2 of dimension n over F are isomorphic, are polynomial-time equivalent.

Now we consider the decision problem of deciding whether a simple algebra A over a number field F is a division algebra. We first note that the set of finite-dimensional simple algebras over F is enumerable; that is, there is a program which takes as input a description of the number field $F = \mathbb{Q}[\alpha]$ and produces a list of (structure constants for) finite-dimensional simple associative algebras over F, with the property that for each finite-dimensional simple algebra A over F, an algebra isomorphic to A is eventually listed. If the decision problem of recognizing division algebras were decidable, we would have a similar program which enumerated all division algebras over F, by considering all finite-dimensional simple algebras and listing only those which were recognized as division algebras.

We also have an algorithm which takes as input a description of a simple algebra A over a number field F and either halts, returning a nonzero zero divisor in A (and hence a proof that A is not a division algebra), or fails to halt. (Consider an algorithm which uses the description of F and basis for A to enumerate the elements of A, until a zero divisor is found.) Now if the decision problem of recognizing division algebras were decidable, we could combine this program with the one described above, using a dovetailing process, to produce a program which always halts, and which either halts after finding a nonzero zero divisor in A, or after finding a division algebra D with the same dimension (n) over F as A such that $A \otimes_F D^* \cong M_{n\times n}(F)$, and proving that A is a division algebra. Thus there is an algorithm for the computational problem of decomposing simple algebras over number fields, and all three of the decision problems we listed are decidable, if and only if the problem of recognizing division algebras over number fields is decidable. To our knowledge, this problem remains open.

We close this section with an even harder problem.

Question 2.5.28. Does there exist an efficiently computable function $U : \mathbb{Z} \to \mathbb{Z}$, such that, if a number field $F = \mathbb{Q}[\alpha]$ and a simple algebra A over F have a description (including minimal polynomial for α and structure constants for A) of (binary) length n, then either A is a division algebra over F, or there exists a description of a nonzero zero divisor in A of (binary) length $U(n)$?

An affirmative answer for this question (and the function U) would yield an algorithm for decomposing simple algebras over number fields, as well as an upper bound on the running time of the algorithm. Proof that a function $U(n) = n^{O(1)}$ exists would also yield a proof that the problem of recognizing division algebras over number fields is in $\mathrm{co} - \mathrm{NP}$.

For a further discussion of the problem of decomposing simple algebras over number fields, and related problems, see Section 7 of [5].

2.6. Related Problems

In this section we mention two sets of problems which are related to the problems (of decomposing associative algebras) of Sections 2.3–2.5. These are proposed as areas for further work.

We first note an alternative decomposition of finite-dimensional associative algebras over a field F. When decomposing an associative algebra A as described in Sections 2.3–2.5, we begin by computing and factoring out the radical of A; we then examine the semi-simple algebra $A/\text{rad}(A)$. While suitable for the problems to be discussed in Section 3, this is inappropriate if we wish to determine whether two associative algebras A and B are isomorphic; it is possible that $A/\text{rad}(A) \cong B/\text{rad}(B)$ while $A \not\cong B$. In fact, there exist structure theorems for finite-dimensional associative algebras over a field F analogous to the structure theorems (2.1.23 and 2.1.25) for semi-simple algebras, which may be more useful for determining isomorphism of arbitrary associative algebras. Proofs of Theorems 2.6.2 and 2.6.3, and further information about the structure of associative algebras which are not necessarily semi-simple, can be found, for example, in Chapter VIII of Curtis and Reiner [31].

Definition 2.6.1. Let A be a finite-dimensional associative algebra over a field F. A two-sided (respectively, left, or right) ideal B of A is *indecomposable* if it is impossible to express B as a direct sum of two nonzero two-sided (respectively, left, or right) ideals.

Note that the condition that an ideal be indecomposable is weaker than the condition that an ideal be irreducible (or minimal): an indecomposable ideal I may contain (as a proper subset) an ideal I_1; however, there will not exist a second ideal I_2 such that $I = I_1 \oplus I_2$. On the other hand, it is clear that any irreducible ideal is also indecomposable.

Theorem 2.6.2. Let A be a finite-dimensional associative algebra over a field F. There exists an integer $s > 0$ and a set of indecomposable two-sided ideals $A_1, A_2, \ldots, A_s$ (the *blocks* of A) such that

$$A = A_1 \oplus A_2 \oplus \cdots \oplus A_s.$$

If $A = B_1 \oplus B_2 \oplus \cdots \oplus B_t$ is any decomposition of A as a direct sum of indecomposable two-sided ideals of A then $s = t$ and (after reordering) $A_i = B_i$ for $1 \le i \le s$.

Thus we have a computational problem analogous to the problem of computing bases for the simple components of a semi-simple algebra — specifically, the problem of computing bases for the blocks of an arbitrary finite-dimensional associative algebra. It is clear that if A and B are finite-dimensional associative algebras with the same dimension over a field F, then $A \cong B$ if and only if these algebras have the same number of blocks, and $A_i \cong B_i$ for $1 \le i \le s$ (for $A = A_1 \oplus \cdots \oplus A_s$ and

$B = B_1 \oplus \cdots \oplus B_s$ the decompositions of A and B into blocks, and after suitable reordering of $B_1, \ldots, B_s$).

There is also a theorem analogous to Theorem 2.1.25.

Theorem 2.6.3. Suppose A is a finite-dimensional associative algebra over a field F; then there exists an integer $l > 0$ and indecomposable left ideals $L_1, L_2, \ldots, L_l$ such that

$$A = L_1 \oplus L_2 \oplus \cdots \oplus L_l;$$

the ideals $L_1, L_2, \ldots, L_l$ are unique up to isomorphism and order of occurrence.

Again, we have an associated computational problem: computing the number l of left ideals in this summation, and computing bases for a set of indecomposable left ideals $L_1, L_2, \ldots, L_l$ whose direct sum is A.

It can be shown that if A is semi-simple then the blocks of A are the same as the simple components of A, and that the indecomposable left ideals of A are also the irreducible left ideals. Thus these computational problems really are generalisations of the problems “Extraction of Simple Components” and (part of) “Decomposition of a Simple Algebra”, of Sections 2.4 and 2.5. Thus we have obvious polynomial-time reductions from these computational problems for semi-simple associative algebras to the corresponding computational problems for arbitrary associative algebras. Given the similarities between the structure theorems for semi-simple associative algebras over F, and Theorems 2.6.2 and 2.6.4, it seems natural to ask whether there are (polynomial-time) reductions in the other direction as well.

Question 2.6.4. Can the problem of computing the blocks (respectively, a set of indecomposable left ideals) of a finite-dimensional associative algebra A over a field F be reduced to the problem of computing the simple components (respectively, a set of irreducible left ideals) of a finite-dimensional semi-simple associative algebra $\hat{A}$ (in particular, of $\hat{A} = A/\text{rad}(A)$)?

We also ask a more general question: How can the decomposition of algebras A and B into blocks, and of blocks into direct sums of indecomposable left ideals, be used to decide whether A is isomorphic to B? Clearly, we gain some information, simply by comparing the dimensions of the blocks and of the indecomposable left ideals of both algebras (just as we gain some information by comparing $A/\text{rad}(A)$ and $B/\text{rad}(B)$); can the decompositions be used to prove that the algebras are isomorphic? In comparison, note that we can decide whether semi-simple algebras over finite fields, $\mathbb{R}$, or $\mathbb{C}$ are isomorphic, by decomposing these algebras using the methods of Sections 2.4–2.5; we can also reduce the problem of deciding whether semi-simple algebras over a number field are isomorphic, to the problem of deciding whether a simple algebra over a number field F is isomorphic to $M_{k\times k}(F)$.

We obtain another important set of algebraic structures over fields F, with a similar set of structure theorems and corresponding computational problems, if we weaken the definition of "algebra".

Definition 2.6.5. A *non-associative algebra* A (that is, "not necessarily associative algebra") over a field F is a vector space over F in which a bilinear composition is defined. For every pair x, $y \in A$ we associate a product $x \circ y \in A$ which satisfies the bilinearity conditions

(i) $(x_1 + x_2) \circ y = x_1 \circ y + x_2 \circ y$, and $x \circ (y_1 + y_2) = x \circ y_1 + x \circ y_2$;

(ii) $\alpha(x \circ y) = (\alpha x) \circ y = x \circ (\alpha y)$, for $\alpha \in F$.

A non-associative algebra A is said to be a *Lie algebra* if its multiplication ($\circ$) also satisfies the Lie conditions

(iii) $x \circ x = 0$, and $(x \circ y) \circ z + (y \circ z) \circ x + (z \circ x) \circ y = 0$.

Example 2.6.6. Let $A = M_{n\times n}(F)$, with addition defined in the usual way, and with the product $x \circ y$ defined by

$$x \circ y = xy - yx,$$

where xy is the product of x and y using (the usual definition of) matrix multiplication. Then it is easily checked that A is a Lie algebra of dimension n^2 over F.

Definition 2.6.7. A *linear Lie algebra* over F is a subspace A of $M_{n\times n}(F)$, with addition and the product $x \circ y$ defined as above.

The structure theory of finite-dimensional Lie algebras is rich; it includes concepts analogous to radicals, semi-simple and simple algebras, and computational problems (for linear Lie algebras) which resemble those discussed in Sections 2.3–2.5, for associative algebras. The structure theory of Lie algebras is discussed in detail by Jacobson [66]; computational problems for Lie algebras (which are beyond the scope of this thesis) are considered in the conference proceedings edited by Beck and Kolman [8], and by Friedl and Rónyai [43].

As was the case for associative algebras over $\mathbb{R}$ or $\mathbb{C}$, Lie algebras over $\mathbb{R}$ and $\mathbb{C}$ (and the problem of decomposing them) have applications in physics and chemistry. Again, the problem of decomposing finite-dimensional Lie algebras over arbitrary fields appears to be difficult (or, at least, is not well understood). Can the methods used here to decompose finite-dimensional associative algebras over $\mathbb{R}$ and $\mathbb{C}$ be applied to the analogous problems for Lie algebras over these fields?

The structure theories of associative algebras, and of Lie algebras, are closely related to the theory of linear representations of groups. The computational aspects of this theory is the subject of Section 3.

3. Computations for Representations of Groups

We now consider algorithms for the construction and decomposition of linear and matrix representations of groups.

We begin, in Section 3.1, by reviewing the definitions of linear and matrix representations of groups over fields. We also recall results about the structure of completely reducible representations of groups, and introduce the problems to be discussed in later sections. The material here is standard; a reader who is familiar with these concepts can skip to later sections.

In the remaining sections, we discuss the problem of decomposing representations of groups over various classes of fields. We start, in Section 3.2, by considering matrix representations of finitely generated groups; we consider problems whose input and output include matrices representing generators of some such group G. It is well known that these problems are closely related to the problems concerning decompositions of finite-dimensional associative algebras which were discussed in Section 2; we use these relationships to obtain polynomial-time (and NC) reductions between problems. Applying the results of Section 2, we obtain polynomial time algorithms for decompositions of matrix representations, and (NC) reductions from these problems to problems concerning factorisation of polynomials. (See, in particular, Theorem 3.2.8 and Corollaries 3.2.9–3.2.11, and Theorem 3.2.17 and Corollary 3.2.18.)

In Section 3.3 we consider the computation of characters of matrix representations, and of character tables for finite groups. We consider a method due to Burnside for the computation of character tables over $\mathbb{C}$, and show that this can be used to computed these tables using polynomial time. We also consider a modification of the basic method, given by Dixon, and show that this can be proved to be asymptotically efficient (and "practical", assuming the extended Riemann hypothesis). We introduce a further modification to obtain an algorithm whose running time can be shown to be bounded by a polynomial function (of the input size) with small degree, without assuming any unproved number theoretic hypotheses. We also use these algorithms to show that the problem of computing the character table (over $\mathbb{C}$) for a finite group is in NC (see Theorem 3.3.27).

Finally, we discuss (in Section 3.4) computational problems related to the representation theory of the symmetric groups, of the general linear group (of nonsingular $n \times n$ matrices over a field F), and of (other) Lie groups. The (few) results we mention here are preliminary. While most of the problems previously discussed have had polynomial-time solutions, the problems mentioned here appear to be much harder. We will see that analysis of the best currently known algorithms prove only that these problems are in PSPACE; that is, they can be solved using deterministic algorithms using only space (memory) which is polynomial in the size of the input. Proving that these problems are hard, or even that the currently used algorithms use superpolynomial time in the worst case, appear to be difficult. We make a modest start here, by showing that a combinatorial method for evaluating characters of the symmetric group computes a superpolynomial number of intermediate values, for infinitely many inputs (see Theorem 3.4.7).

3.1. Linear and Matrix Representations of Groups

We begin with the definitions leading to a structure theory (and associated computational problems) for linear and matrix representations of groups. The material presented here is standard; for a more detailed presentation see, for example, Curtis and Reiner [31]. The reader who is already familiar with this basic material can skip to Section 3.2 (perhaps after noting Definitions 3.1.10–3.1.12, 3.1.14, 3.1.17–3.1.19, and Example 3.1.20).

We will represent elements of a group G by linear transformations, or by nonsingular matrices, over a field F. We begin by reviewing these.

Let M, N be vector spaces over F; then the map $\phi : M \to N$ is an *F-linear transformation* if $\phi(\alpha x + y) = \alpha\phi(x) + \phi(y)$ for all $\alpha \in F$ and all x, $y \in M$. We denote by $\mathrm{Hom}_F(M, N)$ the set of F-linear transformations from M to N. It is clear that if we define the sum $\phi + \psi$ of linear transformations ϕ and ψ so that $(\phi + \psi)(x) = \phi(x) + \psi(x)$ for all $x \in M$, then this sum is also an element of $\mathrm{Hom}_F(M, N)$, and that $\mathrm{Hom}_F(M, N)$ is a commutative group with respect to this addition.

If $M = N$ then the set $\mathrm{Hom}_F(M, N) = \mathrm{Hom}_F(M, M) = \mathrm{End}_F(M)$ is a ring (as well as a commutative group), provided that we define the product $\psi \circ \phi$ of transformations ψ and ϕ according to the rule

$$(\psi \circ \phi)(x) = \psi(\phi(x)) \qquad \text{for all } x \in M$$

(that is, using composition of operators). The identity element of this ring is clearly the transformation 1_M, mapping each element of M to itself; the units are the invertible linear transformations ϕ (those for which there exists a linear transformation ϕ^{-1} such that $\phi^{-1} \circ \phi = \phi \circ \phi^{-1} = 1_M$). We denote the multiplicative group of invertible linear transformations from M to itself by $GL(M, F)$.

Suppose now that the dimensions of M and N are m and n, respectively. We obtain (group and ring) homomorphisms from $\mathrm{Hom}_F(M, N)$, $\mathrm{End}_F(M)$, and $GL(M, F)$ to the additive group $M_{n\times m}(F)$, the ring $M_{m\times m}(F)$, and the multiplicative group $GL(m, F)$ (of nonsingular $m \times m$ matrices with entries in F), respectively, by fixing bases for M and N over F and by mapping each F-linear transformation ϕ to its coefficient matrix with respect to these bases. In particular, suppose x_1, x_2, $\ldots$, x_m and y_1, y_2, $\ldots$, y_n are bases over F for M and N respectively, and let $\phi \in \mathrm{Hom}_F(M, N)$; then there exist constants $\phi_{i\,j}$, for $1 \le i \le n$ and $1 \le j \le m$, such that, for $1 \le j \le m$,

$$\phi(x_j) = \sum_{i=1}^{n} \phi_{i\,j} y_i.$$

We map the linear transformation ϕ to the coefficient matrix $M_\phi \in M_{n\times m}(F)$ whose $(i,\, j)^{\text{th}}$ entry is $\phi_{i\,j}$.

Suppose $M = N$; then if we fix the same basis $x_1,\, x_2,\, \ldots,\, x_m$ for both the domain and range of the linear transformations, we obtain a map from each endomorphism $\phi : M \to M$ to $M_{m\times m}(F)$ which preserves both addition and multiplication: If $\lambda,\, \mu,\, \nu \in \mathrm{End}_F M$ such that $\lambda = \mu \circ \nu$, then it is easily checked that $\lambda(x_j) = \sum_{i=1}^{m} \lambda_{i\,j} x_i$, $\mu(x_j) = \sum_{i=1}^{m} \mu_{i\,j} x_i$, and $\nu(x_j) = \sum_{i=1}^{m} \nu_{i\,j}$ for $\lambda_{i\,j}, \mu_{i\,j}, \nu_{i\,j} \in F$ such that $\lambda_{i\,j} = \sum_{h=1}^{m} \mu_{i\,h} \nu_{h\,j}$ for $1 \le i,\, j \le m$ — so that $M_\lambda = M_\mu M_\nu$, as claimed.

It is easily checked that the identity homomorphism 1_M is mapped to the identity matrix, that these maps are one-to-one and onto, and that the latter map sends invertible linear transformations (only) to nonsingular matrices; we obtain isomorphisms between commutative groups $\mathrm{Hom}_F(M, N)$ and $M_{n\times m}(F)$, between rings $\mathrm{End}_F(M)$ and $M_{m\times m}(F)$, and between groups $GL(M, F)$ and $GL(m, F)$.

We next note that if $x_1,\, x_2,\, \ldots,\, x_m$ and $y_1,\, y_2,\, \ldots,\, y_m$ are two bases for M, and $\phi \in \mathrm{End}_F(M)$, then we obtain two matrices, M_ϕ (with $(i,\, j)^{\text{th}}$ entry $\phi_{i\,j}$) and $\hat{M}_\phi$ (with $(i,\, j)^{\text{th}}$ entry $\hat{\phi}_{i\,j}$) corresponding to ϕ with respect to bases $x_1,\, x_2,\, \ldots,\, x_m$ and $y_1,\, y_2,\, \ldots,\, y_m$ respectively. There exists a linear transformation $\chi \in \mathrm{End}_F(M)$ such that $\chi(x_i) = y_i$ for $1 \le i \le m$. Since $x_1,\, x_2,\, \ldots,\, x_m$ and $y_1,\, y_2,\, \ldots,\, y_m$ are both bases for M it is clear that χ is invertible: $\chi \in GL(M, F)$. Suppose the matrix $X \in GL(m, F)$ (with $(i,\, j)^{\text{th}}$ entry $X_{i\,j}$) corresponds to χ with respect to the basis $x_1,\, x_2,\, \ldots,\, x_m$; that is,

$$y_j = \chi(x_j) = \sum_{i=1}^{m} X_{i\,j} x_i.$$

It is easily checked that $M_\phi X = X\hat{M}_\phi$ — and, since X is invertible, $\hat{M}_\phi = X^{-1} M_\phi X$. Thus, matrices representing the same F-linear transformation with respect to different bases are similar matrices. Changing perspective, we note also that if two linear transformations ϕ and ψ are represented (with respect to different bases for M) by the same matrix M_ϕ, then the linear transformations are *equivalent* in the sense that there exists an invertible linear transformation $\chi \in GL(M, F)$ such that $\psi = \chi^{-1} \circ \phi \circ \chi$.

The "concrete" groups $GL(M, F)$ and $GL(m, F)$ were studied before abstract groups were defined. Since these concrete groups are well understood, and have representations (as groups of matrices) allowing easy implementation of the group operation (via matrix multiplication), it seems natural that these concrete groups have been used to represent more general groups.

Definition 3.1.1. Let G be a group and let M be a finite-dimensional vector space over a field F. A *linear representation* of G over F with *representation space* M is a (group) homomorphism $T : G \to GL(M, F)$. Two linear representations T and T' with representation spaces M and M' respectively are said to be *equivalent* if there exists an F-vector space isomorphism $S : M \to M'$ such that

$$T'(g)S = ST(g) \qquad \text{for all } g \in G.$$

That is,

$$(T'(g) \circ S)(x) = (S \circ T(g))(x) \qquad \text{for all } g \in G \text{ and } x \in M.$$

The dimension of the vector space M over F is called the *degree* of the linear representation $T : G \to GL(M, F)$.

We will discuss linear representations, and provide examples, later. We first present the type of representation to be used more frequently for computation.

Definition 3.1.2. A *matrix representation* of *degree* m of a group G over a field F is a (group) homomorphism $T : G \to GL(m, F)$. Two matrix representations T and T' of G over F are *equivalent* if they have the same degree, m, and there exists a matrix $S \in GL(m, F)$ such that $T'(g) = ST(g)S^{-1}$ for all $g \in G$.

Note that representations are required (only) to be homomorphisms: they are generally neither one-to-one nor onto.

Example 3.1.3. Let $G = \{ g_1, g_2, \ldots, g_n \}$ be a finite group of order $n > 0$. We obtain a matrix representation $\hat{T} : G \to GL(n, F)$, the *regular matrix representation* of $\{ g_1, g_2, \ldots, g_n \}$, as follows. Denote by $e_1, e_2, \ldots, e_n \in M_{n\times 1}(F)$ the column vectors such that e_1 has i^{th} entry 1 and j^{th} entry 0, for $j \neq i$ and $1 \leq i, j \leq n$. We set $\hat{T}(g_i)$ to be the matrix such that

$$\hat{T}(g_i)e_j = e_k \qquad \text{for } k \text{ such that } g_i \cdot g_j = g_k,$$

and for $1 \leq i, j \leq n$. It follows that for $1 \leq i, j, k \leq n$, the $(k, j)^{\text{th}}$ entry of $\hat{T}(g_i)$ is 1 if $g_i \cdot g_j = g_k$, and is 0 otherwise. It is easily checked that $\hat{T}(g_i) \cdot \hat{T}(g_j) = \hat{T}(g_i \cdot g_j)$, and that the identity element of G is mapped to the identity matrix in $GL(n, F)$. It is also clear that $T(g_i)$ is a permutation matrix for $1 \leq i \leq n$, so that the regular matrix representation is also a *permutation representation* of G.

Consider, for example, the cyclic group of order 3, $G = \{ 1, a, a^2 \}$, with $a^3 = 1$. The regular matrix representation $\hat{T} : G \to M_{3\times 3}(F)$ is as follows:

$$\hat{T}(1) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \hat{T}(a) = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \quad \hat{T}(a^2) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}.$$

Recall that we have also defined a "regular matrix representation", $\phi(A)$, for an associative algebra A with respect to a basis $a_1, a_2, \ldots, a_n$ for A over the ground field F (see Definition 2.2.2), and a "group algebra" FG of a finite group $\{ g_1, g_2, \ldots, g_n \}$ with elements of a basis over F in correspondence with the elements of G (see Examples 2.1.6 and 2.2.5). It is easily checked that these "regular matrix representations" coincide: That is, the matrices $\hat{T}(g_i)$ and $\phi(g_i)$ are identical, for $1 \leq i \leq n$.

Example 3.1.4. Any group G has a *trivial representation* $T : G \to GL(n, F)$, with

$$T(g) = \begin{bmatrix} 1 & & & 0 \\ & 1 & & \\ & & \ddots & \\ 0 & & & 1 \end{bmatrix}, \qquad \text{for all } g \in G.$$

We again consider a cyclic group, $C_n = \{ 1, a, a^2, \ldots, a^{n-1} \}$ of order n, with $a^n = 1$.

Example 3.1.5. Suppose F has an n^{th} root of unity, ω; then we have a matrix representation $T : C_n \to GL(1, F)$ with

$$T(a^i) = \begin{bmatrix} \omega^i \end{bmatrix} \qquad \text{for } 0 \leq i < n.$$

If ω is an n^{th} primitive root of unity, then this representation is one-to-one. In particular, if $p - 1$ divides n, and $F = \mathbb{F}_p$, then we can choose any nonzero element of F as an n^{th} root of unity, ω (not necessarily primitive). If $F = \mathbb{C}$, then we can choose $\omega = e^{(2\pi\sqrt{-1})/k}$ for any positive integer k dividing n.

If $T : G \to GL(M, F)$ is a linear representation of a group G, with representation space M of dimension m over F, then we obtain a matrix representation $\hat{T} : G \to GL(m, F)$ by fixing a basis $x_1, x_2, \ldots, x_m$ for M over F, and setting the matrix $\hat{T}(g)$ to be the coefficient matrix for the linear transformation $T(g)$ with respect to this basis. It is clear that this mapping from linear representations of G with representation space M to matrix representations of G of degree m is one-to-one and onto. The relationship between linear representations of G with representation space M and matrix representations of G of degree m is similar to the relationship between invertible linear transformations on M and nonsingular $m \times m$ matrices (discussed at the beginning of this section): We obtain a matrix representation from a linear representation by fixing a basis for M; we obtain equivalent matrix representations (as per Definition 3.1.2) from the same linear representation by choosing different bases for M; and linear representations which correspond to the same matrix representation (with respect to different bases) are equivalent (as per Definition 3.1.1).

The notion of a representation space, and of a linear representation of a group, is important; it is useful in applying representation theory to other domains, and simplifies some of the definitions which follow. On the other hand, we are interested in computational problems, which require "concrete" (matrix) representations as input and output. Hence we will use both notions of "representation" of a group, stressing matrix representations when discussing computational problems.

We next note several ways of generating new matrix representations from given representations.

Example 3.1.6. Suppose $T_1 : G \to GL(N_1, F)$ and $T_2 : G \to GL(N_2, F)$ are two linear representations of a group G, of degrees n_1 and n_2 respectively. We obtain a new linear representation $T = T_1 \oplus T_2$, the *direct sum* of T_1 and T_2, by defining the map

$$T : G \to GL(N_1 \oplus N_2, F)$$

so that

$$T(g)(x + y) = T_1(g)(x) + T_2(g)(y) \qquad \text{for all } g \in G, x \in N_1, \text{ and } y \in N_2.$$

It is easily checked that T maps the identity element of G to the identity map on $N_1 \oplus N_2$ and that $T(g_1 \cdot g_2) = T(g_1) \circ T(g_2)$ for all g_1, $g_2 \in G$.

Now if we fix bases x_1, x_2, $\ldots$, x_{n_1} and y_1, y_2, $\ldots$, y_{n_2} for spaces N_1 and N_2 respectively, we obtain matrix representations $\hat{T}_1 : G \to GL(n_1, F)$ corresponding to T_1, and $\hat{T}_2 : G \to GL(n_2, F)$ corresponding to T_2, with respect to these bases. If we set x_1, x_2, $\ldots$, x_{n_1}, y_1, y_2, $\ldots$, y_{n_2} as our basis for $N_1 \oplus N_2$ then the matrix representation $\hat{T} = \hat{T}_1 \oplus \hat{T}_2$ with respect to this basis sends each element of G to a block diagonal matrix,

$$\hat{T}(g) = \begin{bmatrix} \hat{T}_1(g) & 0 \\ 0 & \hat{T}_2(g) \end{bmatrix} \qquad \text{for all } g \in G.$$

This is easily verified, using the relation $T_1(g)(x_i) = \sum_{j=1}^{n_1} (\hat{T}_1(g))_{j\,i} x_j$, the relation $T_2(g)(y_k) = \sum_{l=1}^{n_2} (\hat{T}_2(g))_{l\,k} y_l$, and the above definition of $T_1 \oplus T_2$.

Example 3.1.7. Suppose again that $T_1 : G \to GL(N_1, F)$ and $T_2 : G \to GL(N_2, F)$ are two linear representations of a group G, of degrees n_1 and n_2 respectively. By Proposition 2.2.12, the tensor product $N_1 \otimes_F N_2$ is a vector space of dimension $n_1 n_2$ over F. We obtain a "tensor representation of G",

$$T_1 \otimes_F T_2 : G \to GL(N_1 \otimes_F N_2, F),$$

by defining the action of a group element g on elements of the form $x \otimes_F y$ (for $x \in N_1$ and $y \in N_2$) componentwise, and using linearity to extend this to arbitrary elements of $N_1 \otimes_F N_2$. That is, for all $g \in G$, $x \in N_1$, and $y \in N_2$,

$$(T_1 \otimes_F T_2)(g)(x \otimes_F y) = ((T_1(g)(x)) \otimes_F (T_2(g)(y))),$$

and for $r > 0$, $x_1, x_2, \ldots, x_r \in N_1$, $y_1, y_2, \ldots, y_r \in N_2$, and $g \in G$,

$$(T_1 \otimes_F T_2)(g) \left(\sum_{i=1}^{r} (x_i \otimes_F y_i) \right) = \sum_{i=1}^{r} ((T_1 \otimes_F T_2)(g)(x_i \otimes_F y_i)).$$

This defines a (unique) group homomorphism from G to $GL(N_1 \otimes_F N_2, F)$ — and, hence, a linear representation of G. (See, for example, Section 12A of [31] for a proof that this map is unique and well defined.)

Suppose now that $\hat{T}_1 : G \to GL(n_1, F)$ and $\hat{T}_2 : G \to GL(n_2, F)$ are matrix representations corresponding to the the linear representations T_1 and T_2 with respect to bases $x_1, x_2, \ldots, x_{n_1}$ and $y_1, y_2, \ldots, y_{n_2}$ for N_1 and N_2 respectively. We obtain a matrix representation $\hat{T} = \hat{T}_1 \otimes_F \hat{T}_2 : G \to GL(n_1 n_2, F)$ from the linear representation $T = T_1 \otimes_F T_2$, using the basis

$$x_1 y_1,\ x_2 y_1,\ \ldots,\ x_{n_1} y_1,\ x_1 y_2,\ \ldots,\ x_{n_1} y_{n_2 - 1},\ x_1 y_{n_2},\ x_2 y_{n_2},\ \ldots,\ x_{n_1} y_{n_2}$$

for $N_1 \otimes_F N_2$ over F. Suppose the matrix $\hat{T}_2(g)$ has $(i,\, j)^{\text{th}}$ entry $\beta_{i\,j}$, for $1 \le i,\, j \le n_2$; then the matrix $\hat{T}(g) = (\hat{T}_1 \otimes_F \hat{T}_2)(g)$ has the form

$$(\hat{T}_1 \otimes_F \hat{T}_2)(g) = \begin{bmatrix} \beta_{1\,1}\hat{T}_1(g) & \beta_{1\,2}\hat{T}_1(g) & \cdots & \beta_{1\,n_2}\hat{T}_1(g) \\ \beta_{2\,1}\hat{T}_1(g) & \beta_{2\,2}\hat{T}_1(g) & \cdots & \beta_{2\,n_2}\hat{T}_1(g) \\ \vdots & \vdots & \ddots & \vdots \\ \beta_{n_2\,1}\hat{T}_1(g) & \beta_{n_2\,2}\hat{T}_1(g) & \cdots & \beta_{n_2\,n_2}\hat{T}_1(g) \end{bmatrix}.$$

That is, the matrix $(\hat{T}_1 \otimes_F \hat{T}_2)(g)$ is obtained from $\hat{T}_2(g)$ by replacing each entry $\beta_{i\,j}$ by the block matrix $\beta_{i\,j}\hat{T}_1(g)$.

In the next example we note that the tensor product construction can be used to change the ground field for a representation — that is, to perform "extension of scalars".

Example 3.1.8. Let $T : G \to GL(N, F)$ be a linear representation of a group G of degree n over a field F. Let E be an extension field of F; then we can view E as a vector space over F. Let $1_E : G \to GL(E, F)$ be the "trivial" representation of G — so that

$$1_E(g)(x) = x \quad \text{for all } g \in G \text{ and } x \in E$$

(*cf.* Example 3.1.4, for matrix representations). As in Example 3.1.7 we form a "product" representation

$$T \otimes_F 1_E : G \to GL(N \otimes_F E, F).$$

It is easily checked that for each $g \in G$, the linear transformation $T \otimes_F 1_E(g)$ is an E-linear transformation, as well as F-linear. That is, $T \otimes_F 1_E(g) \in GL(N \otimes_F E, E)$. We thus obtain a linear transformation of G of degree n over E,

$$T_E : G \to GL(N \otimes_F E, E),$$

by setting $T_E(g)$ to be $T \otimes_F 1_E(g)$, viewed now as an element of $GL(N \otimes_F E, E)$.

If $\hat{T} : G \to GL(n, F)$ is a matrix representation of G of degree n over F then we obtain a matrix representation $\hat{T}_E : G \to GL(n, E)$ by the same method. It is easily checked that the matrices $\hat{T}_E(g)$ and $\hat{T}(g)$ are actually *identical* for all $g \in G$: We have simply used the obvious embedding of $M_{n\times n}(F)$ in $M_{n\times n}(E)$.

We can use this process to obtain from a linear representation T over F (or a matrix representation $\hat{T}$) a linear representation T_E over E (or matrix representation $\hat{T}_E$), even when E is infinite-dimensional over F. In later sections we will consider linear representations $T_{\mathbb{R}}$ and $T_{\mathbb{C}}$ (and matrix representations $\hat{T}_{\mathbb{C}}$ and $\hat{T}_{\mathbb{R}}$) obtained from representations T (or $\hat{T}$) over number fields.

We are interested in representations which cannot be expressed as direct sums of smaller representations, and in expressing "general" representations as direct sums of these "irreducible" representations.

Definition 3.1.9. Let $T : G \to GL(M, F)$ be a linear representation of a group G over a field F. A subspace N of M is a *G-subspace* of M if

$$T(g)x \in N \qquad \text{for all } g \in G \text{ and } x \in N.$$

That is, N is invariant with respect to each linear transformation $T(g)$ for $g \in G$.

If T, N, and M are as above and we define

$$T_1(g) = T(g)|_N \qquad \text{for } g \in G,$$

where $T(g)|_N$ denotes the restriction of $T(g)$ to N, then T_1 is a linear representation of G over F with representation space N.

Definition 3.1.10. A linear representation $T : G \to GL(M, F)$ with nonzero representation space M is an *irreducible* linear representation of G if the only G-subspaces of M are (0) and M; otherwise, T is a *reducible* linear representation.

We also have a weaker condition than irreducibility.

Definition 3.1.11. A linear representation $T : G \to GL(M, F)$ with nonzero representation space M is an *indecomposable* linear representation if there do not exist proper G-subspaces M_1 and M_2 such that $M = M_1 \oplus M_2$; otherwise, T is a *decomposable* linear representation.

Definition 3.1.12. A matrix representation $\hat{T} : G \to GL(m, F)$ is *irreducible* (respectively, *indecomposable*) if there exists a linear representation $T : G \to GL(M, F)$ such that $\hat{T}$ corresponds to T (with respect to some basis for M), and T is an irreducible (respectively, indecomposable) linear representation. Otherwise, T is *reducible* (respectively, *decomposable*).

Example 3.1.13. Let $G = C_p = \{ 1, a, a^2, \ldots, a^{p-1} \}$ (with $a^p = 1$) for prime $p > 0$. Let $F = \mathbb{F}_p$, and suppose M is an F-vector space of dimension 2, with basis m_1, m_2 over F. We define a map $T : G \to GL(M, F)$ by setting

$$T(a)(m_1) = m_1 \qquad \text{and} \qquad T(a)(m_2) = m_1 + m_2,$$

and using additivity to extend this map to arbitrary elements of M. Then,

$$T(a^i)(m_1) = m_1 \qquad \text{and} \qquad T(a^i)(m_2) = im_1 + m_2,$$

and $T(a^i)(\lambda_1 m_1 + \lambda_2 m_2) = (\lambda_1 + i\lambda_2)m_1 + \lambda_2 m_2$ for $i \geq 0$ and $\lambda_1, \lambda_2 \in F$. Since $T(a^p)(m_1) = T(a^0)(m_1) = m_1$ and $T(a^p)(m_2) = T(a^0)(m_2) = m_2$, it is clear that T is a linear representation of G over F, with representation space M.

It is also clear that the one-dimensional subspace $M_1 = \{ \lambda_1 m_1 \ : \ \lambda_1 \in F \}$ of M is a G-subspace of M. The restriction of T to M_1, $T_1 : G \to GL(M_1, F)$, is a linear representation of G (a *trivial* representation, as per Example 3.1.4).

Fixing the basis $\{ m_1, m_2 \}$ for M over F, we obtain from T the matrix representation $\hat{T} : G \to GL(2, F)$ with

$$\hat{T}(a) = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}, \qquad \text{and} \qquad \hat{T}(a^i) = \begin{bmatrix} 1 & i \\ 0 & 1 \end{bmatrix} \ \text{ for } i \geq 0.$$

Fixing the basis $\{ m_1 \}$ for M_1 over F, we obtain from T_1 the matrix representation $\hat{T}_1 : G \to GL(1, F)$ with

$$\hat{T}(a) = \hat{T}(a^i) = [1] \text{ for } i \geq 0.$$

We have showed that T is reducible. T is also indecomposable — for if there existed G-subspaces N_1 and N_2 such that $M = N_1 \oplus N_2$, then it would follow that N_1 and N_2 were both invariant with respect to $T(a)$. Since M has dimension 2, it would also follow that N_1 and N_2 have dimension 1, so that each would contain an eigenvalue for $T(a)$ — contradicting the fact that $\hat{T}(a)$ is not similar to a diagonal matrix. Thus T is indecomposable, but not irreducible.

As noted in the above example, there exist linear representations which are indecomposable, but not irreducible; on the other hand, it is clear from the definitions that every irreducible representation is indecomposable.

The example also illustrates a general property of reducible linear and matrix representations: Suppose $T : G \to GL(N, F)$ is a linear representation and that N_1 is a proper G-subspace of N. Then there exists a subspace N_2 of N (not necessarily a G-subspace) such that $N = N_1 \oplus N_2$. Now, for all $x \in N_1$ and $y \in N_2$, and for all $g \in G$,

$$T(g)(x) \in N_1 \qquad \text{and} \qquad T(g)(y) \in N_1 \oplus N_2 = N.$$

If $x_1, x_2, \ldots, x_{n_1}$ is a basis for N_1 over F, and $y_1, y_2, \ldots, y_{n_2}$ is a basis for N_2 over F, then we have a basis $x_1, x_2, \ldots, x_{n_1}, y_1, y_2, \ldots, y_{n_2}$ for N, and a matrix representation $\hat{T} : G \to GL(n, F)$ corresponding to this basis, so that for all $g \in G$,

$$\hat{T}(g) = \begin{bmatrix} \hat{T}_1(g) & U(g) \\ 0 & \hat{T}_2(g) \end{bmatrix}.$$

Here, $\hat{T}_1$ is the matrix representation corresponding to the restriction T_1 of T to N_1; $\hat{T}_2$ is multiplicative, and maps the identity element of G to the identity matrix of order n_2, so $\hat{T}_2$ is also a matrix representation (which does *not* generally correspond to the restriction of T to a G-subspace of N). The off-diagonal matrix, $U(g)$, is not generally multiplicative (as a function of $g \in G$); $U(g) = 0$ for all $g \in G$, so that $\hat{T}(g)$ is block diagonal, if and only if N_2 is also a G-subspace; then $\hat{T}_2(g)$ is the matrix representation corresponding to the basis $y_1, y_2, \ldots, y_{n_2}$ for N_2 and the restriction of the linear representation T to this subspace.

We are interested in the special case that $U(g) = 0$ for all $g \in G$, so that T is decomposable as well as reducible.

Definition 3.1.14. A linear representation $T : G \to GL(N, F)$ is *completely reducible* if, for every G-subspace N_1 of N, there exists another G-subspace N_2 of N such that $N = N_1 \oplus N_2$ (as vector spaces over F). A matrix representation $\hat{T}$ is *completely reducible* if it corresponds to a completely reducible linear representation with respect to some basis for the representation space.

Any irreducible (linear or matrix) representation is (trivially) completely reducible. If a representation T is completely reducible, then T is indecomposable if and only if T is irreducible. Hence the representation T of Example 3.1.13 is not completely reducible.

Completely reducible representations have a useful characterisation, in terms of irreducible representations.

Theorem 3.1.15. Let $T : G \to GL(M, F)$ be a completely reducible representation; then M is a direct sum of G-subspaces M_1, M_2, …, M_k for some $k > 0$, such that the restriction $T_i : G \to GL(M_i, F)$ of T to M_i is an irreducible linear representation of G, for $1 \leq i \leq k$.

Corollary 3.1.16. Let $\hat{T} : G \to GL(m, F)$ be a completely reducible matrix representation of G over F; then $\hat{T}$ is equivalent to a direct sum of irreducible matrix representations $\hat{T}_i : G \to GL(m_i, F)$ for $1 \leq i \leq k$, for some integer $k > 0$.

For a proof of Theorem 3.1.15 see, for example, Curtis and Reiner [31] (stated there as Theorem 10.7). It is clear that the converses of the above theorem and corollary are also true: Every direct sum of irreducible representations is completely reducible.

We will consider the decision problem of identifying completely reducible representations, and the computational problem of expressing them as direct sums of irreducible representations, for classes of groups for which representations can be specified easily. We will also consider an intermediate type of representation (between "completely reducible" and "irreducible").

Definition 3.1.17. A linear (respectively, matrix) representation of a group G over a field F is *isotypic* if it is equivalent to a direct sum of equivalent irreducible linear (respectively, matrix) representations of G over F.

Clearly, every irreducible representation is isotypic, and every isotypic representation is completely reducible. We will consider the decomposition of completely reducible representations in two stages, as described below.

Definition 3.1.18. A linear representation $T_1 : G \to GL(M_1, F)$ is a *component* of a linear representation $T : G \to GL(M, F)$ if M_1 is a G-subspace, T_1 is the restriction of T to M_1, and there exists a G-subspace M_2 of M such that $M = M_1 \oplus M_2$ (so that T_1 is a *direct summand* of T).

A matrix representation $\hat{T}_1$ of G is a *component* of a matrix representation $\hat{T}$ if $\hat{T}$ is equivalent to $\hat{T}_1 \oplus \hat{T}_2$ for some matrix representation $\hat{T}_2$.

Definition 3.1.19. Let $T : G \to GL(M, F)$ be a linear representation of G. A set of linear representations $T_i : G \to GL(M_i, F)$, for $1 \leq i \leq k$, is a set of *isotypic components* of T if $M = M_1 \oplus M_2 \oplus \cdots \oplus M_k$, T_i is the restriction of T to the nonzero

G-subspace M_i, and T_i is isotypic, for $1 \le i \le k$, and if no nonzero component of T_i is equivalent to a component of T_j for $1 \le i, j \le k$ with $i \ne j$ (that is, if $T_i \oplus T_j$ is not isotypic).

A set of matrix representations $\hat{T}_1, \hat{T}_2, \ldots, \hat{T}_k$ of G over F is a set of *isotypic components* of a matrix representation $\hat{T}$ if each has positive degree, $\hat{T}$ is equivalent to the direct sum $\hat{T}_1 \oplus \hat{T}_2 \oplus \cdots \oplus \hat{T}_k$, $\hat{T}_i$ is isotypic for $1 \le i \le k$, and $\hat{T}_i \oplus \hat{T}_j$ is not isotypic for $1 \le i, j \le k$ with $i \ne j$.

As a first stage of the decomposition of a completely reducible (linear or matrix) representation T of G, we will compute a set of isotypic components for T. As a second (final) stage, we compute a set of irreducible components for each of these isotypic representations.

Example 3.1.20. Let $F = \mathbb{C}$, $G = D_3$, the dihedral group of order 6 (with generators a and b, such that $a^3 = b^2 = 1$ and $ab = ba^2$), and let $\hat{T} : G \to GL(6, F)$ be the regular matrix representation for $G = \{\, 1, a, a^2, b, ab, a^2b \,\}$. Then

$$\hat{T}(a) = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}, \qquad \hat{T}(b) = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix},$$

and $\hat{T}(g)$ can be computed for all other elements g of G using the multiplicativity of $\hat{T}$.

Let ω be a 3rd primitive root of unity in $\mathbb{C}$, so $\omega^2 + \omega + 1 = 0$. Consider the matrix $X \in GL(6, F)$,

$$X = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & -1 \\ 1 & 1 & \omega & \omega^2 & \omega & -\omega^2 \\ 1 & 1 & \omega^2 & \omega & \omega^2 & -\omega \\ 1 & -1 & 1 & 1 & -1 & 1 \\ 1 & -1 & \omega & \omega^2 & -\omega & \omega^2 \\ 1 & -1 & \omega^2 & \omega & -\omega^2 & \omega \end{bmatrix}.$$

X is nonsingular, with inverse

$$X^{-1} = \frac{1}{6} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & -1 & -1 & -1 \\ 1 & \omega^2 & \omega & 1 & \omega^2 & \omega \\ 1 & \omega & \omega^2 & 1 & \omega & \omega^2 \\ 1 & \omega^2 & \omega & -1 & -\omega^2 & -\omega \\ -1 & -\omega & -\omega^2 & 1 & \omega & \omega^2 \end{bmatrix}.$$

$\hat{T}$ is equivalent to the matrix representation $\hat{U} : G \to GL(6, F)$ with $\hat{U}(g) = X^{-1}\hat{T}(g)X$ for all $g \in G$; now it is easily checked that

$$\hat{U}(a) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & \omega^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & \omega & 0 & 0 \\ 0 & 0 & 0 & 0 & \omega^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & \omega \end{bmatrix}, \qquad \hat{U}(b) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}.$$

Since a and b generate G, it follows that $\hat{U}(g)$ is block diagonal, with two (upper) 1×1 blocks and two (lower) 2×2 blocks, for all $g \in G$. Thus $\hat{U}$ (and hence $\hat{T}$) has components $\hat{U}_1$, $\hat{U}_2$, $\hat{U}_3$, and $\hat{U}_4$ such that

$$\hat{U} = \hat{U}_1 \oplus \hat{U}_2 \oplus \hat{U}_3 \oplus \hat{U}_4,$$

with $\hat{U}_1$, $\hat{U}_2 : G \to GL(1, F)$, $\hat{U}_3$, $\hat{U}_4 : G \to GL(2, F)$,

$$\hat{U}_1(a) = [1], \quad \hat{U}_1(b) = [1], \qquad \hat{U}_2(a) = [1], \quad \hat{U}_2(b) = [-1],$$

and

$$\hat{U}_3(a) = \hat{U}_4(a) = \begin{bmatrix} \omega^2 & 0 \\ 0 & \omega \end{bmatrix}, \qquad \hat{U}_3(b) = \hat{U}_4(b) = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}.$$

Now we note that $\hat{U}_3$ and $\hat{U}_4$ are each irreducible; for if λ_1, $\lambda_2 \in F$ with at least one nonzero, then it is easily checked that $\begin{bmatrix} 1 \\ 0 \end{bmatrix}$ and $\begin{bmatrix} 0 \\ 1 \end{bmatrix}$ both belong to the subspace of $M_{2\times1}(F)$ spanned by the vectors $\begin{bmatrix} \lambda_1 \\ \lambda_2 \end{bmatrix}$, $\hat{U}_3(a) \begin{bmatrix} \lambda_1 \\ \lambda_2 \end{bmatrix}$, and $\hat{U}_3(b) \begin{bmatrix} \lambda_1 \\ \lambda_2 \end{bmatrix}$; since $\hat{U}_3 = \hat{U}_4$, this is also true when $\hat{U}_3$ is replaced by $\hat{U}_4$. Since $\hat{U}_3$ and $\hat{U}_4$ are equal, they are (trivially) equivalent. $\hat{U}_1$ and $\hat{U}_2$ are clearly not equivalent, since matrices $\hat{U}_1(b)$ and $\hat{U}_2(b)$ are not similar.

Thus the matrix representation $\hat{T}$ has a set of isotypic components $\hat{T}_1$, $\hat{T}_2$, $\hat{T}_3$, such that $\hat{T}_1$, $\hat{T}_2 : G \to GL(1, F)$, $\hat{T}_3 : G \to GL(4, F)$, and

$$\begin{aligned} \hat{T}_1(a) &= [1], & \hat{T}_1(b) &= [1], \\ \hat{T}_2(a) &= [1], & \hat{T}_2(b) &= [-1], \end{aligned}$$

$$\hat{T}_3(a) = \begin{bmatrix} \omega^2 & 0 & 0 & 0 \\ 0 & \omega & 0 & 0 \\ 0 & 0 & \omega^2 & 0 \\ 0 & 0 & 0 & \omega \end{bmatrix}, \qquad \hat{T}_3(b) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}.$$

Representations $\hat{T}_1$ and $\hat{T}_2$ are irreducible, while $\hat{T}_3 = \hat{T}_{3\,1} \oplus \hat{T}_{3\,2}$ for equivalent (in fact, identical) irreducible representations $\hat{T}_{3\,1}$, $\hat{T}_{3\,2} : G \to GL(2, F)$ with

$$\hat{T}_{3\,1}(a) = \hat{T}_{3\,2}(a) = \begin{bmatrix} \omega^2 & 0 \\ 0 & \omega \end{bmatrix}, \qquad \hat{T}_{3\,1}(b) = \hat{T}_{3\,2}(b) = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}.$$

In the next three sections we consider the above computational problems, and the problem of deciding whether two representations of a group are equivalent, for various classes of groups. While we have a natural “description” of a matrix representation of a finitely generated group over a field — namely, the matrices representing generators for the group — we will see that different (and often more concise) descriptions are used when providing input and output for problems. Consequently, we will give more detailed definitions of these problems in later sections.

3.2. Matrix Representations of Finitely Generated Groups

In this section we restrict attention to representations of finitely generated groups. We specify a matrix representation $\hat{T} : G \to GL(m, F)$ for such a group G by giving the matrices $\hat{T}(g_1)$, $\hat{T}(g_2)$, $\ldots$, $\hat{T}(g_n)$, for a set of generators g_1, g_2, $\ldots$, g_n of G. Hence, we consider problems for which the only source of information about the group G is its image, $\hat{T}(G)$.

We start with formal definitions of the problems introduced at the end of Section 3.1.

Problem **Identification of a Completely Reducible Representation**

Input.
- Integers n, $m > 0$.
- Matrices τ_1, τ_2, $\ldots$, $\tau_n \in GL(m, F)$ such that $\tau_j = \hat{T}(g_j)$ for $1 \le j \le n$, for a finitely generated group G with generators g_1, g_2, $\ldots$, g_n, and such that $\hat{T} : G \to GL(m, F)$ is a matrix representation of G over the field F.

Output.
true if $\hat{T}$ is a completely reducible matrix representation of G;
false otherwise.

Problem **Isotypic Components of a Completely Reducible Representation**

Input.
- Integers n, $m > 0$.
- Matrices τ_1, τ_2, $\ldots$, $\tau_n \in GL(m, F)$ such that $\tau_j = \hat{T}(g_j)$ for $1 \le j \le n$, for a finitely generated group G with generators g_1, g_2, $\ldots$, g_n, and such that $\hat{T} : G \to GL(m, F)$ is a completely reducible matrix representation of G over the field F.

Output.
- Integers k, $m_1, m_2, \ldots, m_k > 0$, such that $m_1 + m_2 + \cdots + m_k = m$.
- Matrices $\tau_{i\,1}$, $\tau_{i\,2}$, $\ldots$, $\tau_{i\,n} \in GL(m_i, F)$, for $1 \le i \le k$, such that $\tau_{i\,j} = \hat{T}_i(g_j)$ for $1 \le j \le n$, and the isotypic matrix representations $\hat{T}_1$, $\hat{T}_2$, $\ldots$, $\hat{T}_k$ form a set of isotypic components for $\hat{T}$.
- Matrices X and X^{-1} in $GL(m, F)$ such that $\hat{T}(g_j) = X^{-1}\mathrm{Diag}(\hat{T}_1(g_j), \hat{T}_2(g_j), \ldots, \hat{T}_k(g_j))X \quad \text{for } 1 \le j \le n.$

Problem **Irreducible Components of an Isotypic Representation**

Input.
- Integers n, $m > 0$.
- Matrices $\tau_1, \tau_2, \ldots, \tau_n \in GL(m, F)$ such that $\tau_j = \hat{T}(g_j)$ for $1 \leq j \leq n$, for a finitely generated group G with generators $g_1, g_2, \ldots, g_n$, and such that $\hat{T} : G \to GL(m, F)$ is an isotypic matrix representation of G over the field F.

Output.
- Integers k, $l > 0$ such that $k \cdot l = m$.
- Matrices $\tau_{i\,1}, \tau_{i\,2}, \ldots, \tau_{i\,n} \in GL(l, F)$, for $1 \leq i \leq k$, such that $\tau_{i\,j} = \hat{T}_i(g_j)$ for $1 \leq j \leq n$, and the equivalent irreducible matrix representations $\hat{T}_1, \hat{T}_2, \ldots, \hat{T}_k$ form a set of irreducible components for $\hat{T}$.
- Matrices X and X^{-1} in $GL(m, F)$ such that $\hat{T}(g_j) = X^{-1}\mathrm{Diag}(\hat{T}_1(g_j), \hat{T}_2(g_j), \ldots, \hat{T}_k(g_j))X \quad$ for $1 \leq j \leq n$.

Problem **Equivalence of Representations**

Input.
- Integers $n, m > 0$.
- Matrices $\tau_{1\,1}, \tau_{1\,2}, \ldots, \tau_{1\,n}$ and $\tau_{2\,1}, \tau_{2\,2}, \ldots, \tau_{2\,n} \in GL(m, F)$ such that $\tau_{1\,j} = \hat{T}_1(g_j)$ and $\tau_{2\,j} = \hat{T}_2(g_j)$ for $1 \leq j \leq n$, for a finitely generated group G with generators $g_1, g_2, \ldots, g_n$, and such that $\hat{T}_1 : G \to GL(m, F)$ and $\hat{T}_2 : G \to GL(m, F)$ are matrix representations of the group G over the field F.

Output. Matrix $X \in M_{m \times m}(F)$ such that
- X is nonsingular, and $\hat{T}_2(g_j) = X^{-1}\hat{T}_1(g_j)X$ for $1 \leq j \leq n$, if the representations $\hat{T}_1$ and $\hat{T}_2$ are equivalent;
- $X = 0$, if the representations are not equivalent.

We do not insist that a set of generators for G be minimal. In particular, we allow $g_1, g_2, \ldots, g_n$ as a set of generators for a finite group $G = \{ g_1, g_2, \ldots, g_n \}$ with n elements.

As in Section 2, we are interested both in arithmetic algorithms for problems over arbitrary fields F, and in Boolean algorithms for problems over concrete fields (finite fields, number fields, $\mathbb{R}$, and $\mathbb{C}$). For representations over $\mathbb{R}$ or $\mathbb{C}$ we assume when considering Boolean algorithms that our inputs, the entries of the matrices $\hat{T}(g_j)$ for $1 \leq j \leq n$, all lie in some number field F (see Example 3.1.8). As in Section 2, we will be able to compute a decomposition over $\mathbb{R}$ or $\mathbb{C}$ by working in some finite algebraic extension E over F, so that we can use exact representations of all the values to be computed.

The main results of this section are reductions between each of the first three of the above problems and the problems discussed in Sections 2.3–2.5. These reductions, and some of their consequences, are presented in Section 3.2.1. We obtain quite different results when we consider the fourth problem given above (and the corresponding problem for associative algebras), in Section 3.2.2.

3.2.1. Reductions Between Problems

We now present the main results of Section 3.2 — reductions between each of the first three problems defined above, and the problems for associative algebras. The reductions are useful for either sequential or parallel computations. We state them in two parts: Reductions from problems for representations of groups to problems for associative algebras are given in Theorem 3.2.8, while reductions in the other direction are given in Theorem 3.2.17. Some consequences of these reductions are noted in Corollaries 3.2.9–3.2.11 and 3.2.18.

We begin with some well known relationships between representations of groups and associative algebras. These will be used to prove that our reductions are correct.

Suppose now that M is a finite-dimensional vector space over a field F; then $\mathrm{Hom}_F(M, M)$ is also a finite-dimensional vector space over F, and forms an associative algebra A over F, using composition of linear transformations as multiplication in A. If, in addition, $T : G \to GL(M, F)$ is a linear representation of a group G, then the set of all finite F-linear combinations of linear transformations $T(g)$ for $g \in G$ (that is, the F-space spanned by the set $\{ T(g) : g \in G \}$) forms an F-subspace of $A = \mathrm{Hom}_F(M, M)$ which contains the multiplicative identity of A and is closed under multiplication. Thus, this set forms an F-subalgebra of $\mathrm{Hom}_F(M, M)$. Similarly, if $\hat{T} : G \to GL(m, F)$ is a matrix representation of F, then we obtain a matrix algebra (which is an F-subalgebra of $M_{m\times m}(F)$) by considering finite linear combinations of the images $\hat{T}(g)$ of elements g of G.

Definition 3.2.1. Let $T : G \to GL(M, F)$ be a linear representation of a group G over a field F. The *enveloping algebra* of T, $\mathrm{env}(T)$, is the smallest (linear) F-subalgebra of $\mathrm{Hom}_F(M, M)$ containing every F-linear transformation $T(g)$ for $g \in G$.

Let $\hat{T} : G \to GL(m, F)$ be a matrix representation of G over F. The *enveloping algebra* of $\hat{T}$, $\mathrm{env}(\hat{T})$, is the smallest (matrix) F-subalgebra of $M_{m\times m}(F)$ containing every matrix $\hat{T}(g)$ for $g \in G$.

Suppose again that $T : G \to GL(M, F)$ is a linear representation of a group G over F, and let $a \in F$ and ϕ, $\psi \in \mathrm{Hom}_F(M, M)$ such that ϕ and ψ commute with $T(g)$

for all $g \in G$: $\phi \circ T(g) = T(g) \circ \phi$ and $\psi \circ T(g) = T(g) \circ \psi$ for all $g \in G$. It is clear that $a\phi + \psi$ also commutes with each transformation $T(g)$, as does $\phi \circ \psi$: The set of linear transformations in $\mathrm{Hom}_F(M, M)$ commuting with every transformation $T(g)$ forms an F-subalgebra of $\mathrm{Hom}_F(M, M)$. Similarly, the set of matrices in $M_{m\times m}(F)$ commuting with every matrix $\hat{T}(g)$ for $g \in G$, for a matrix representation $\hat{T}$ of G, forms an F-subalgebra of $M_{m\times m}(F)$.

Definition 3.2.2. Let $T : G \to GL(M, F)$ be a linear representation of a group G over a field F. The *commutator algebra* of T, $\mathrm{com}(T)$, is the algebra over F consisting of all linear transformations in $\mathrm{Hom}_F(M, M)$ which commute with every transformation $T(g)$ for $g \in G$.

Let $\hat{T} : G \to GL(m, F)$ be a matrix representation of a group G over a field F. The *commutator algebra* of $\hat{T}$, $\mathrm{com}(\hat{T})$, is the (matrix) algebra over F consisting of all matrices in $M_{m\times m}(F)$ which commute with every matrix $\hat{T}(g)$ for $g \in G$.

We will relate the decomposition of the linear representation $T : G \to GL(M, F)$ (respectively, the matrix representation $\hat{T} : G \to GL(m, F)$) to the decomposition of the associative algebras $\mathrm{env}(T)$ and $\mathrm{com}(T)$ (respectively, of the associative matrix algebras $\mathrm{env}(\hat{T})$ and $\mathrm{com}(\hat{T})$).

Proposition 3.2.3. Let $T : G \to GL(M, F)$ be a linear representation of a group G.

(a) T is completely reducible if and only if $\mathrm{env}(T)$ is semi-simple.

(b) If T is completely reducible then $\mathrm{com}(T)$ is semi-simple.

The converse of Theorem 3.2.3(b) is not generally true (see Example 3.2.14 for a counterexample).

Proposition 3.2.4. Let $T : G \to GL(M, F)$ be a completely reducible linear representation of a group G.

(a) T is isotypic if and only if $\mathrm{env}(T)$ is simple.

(b) T is isotypic if and only if $\mathrm{com}(T)$ is simple.

(c) If the semi-simple algebra $\mathrm{env}(T)$ has a set of central primitive idempotents $e_1, e_2, \ldots, e_k$ (so $e_1 + e_2 + \cdots + e_k = 1$ and $e_i e_j = \delta_{i\,j} e_i$ for $1 \le i, j \le k$), then T has isotypic components $T_1, T_2, \ldots, T_k$, for $T_i = T|_{M_i} : G \to GL(M_i, F)$ and for $M_i = e_i(M)$, for $1 \le i \le k$.

Proposition 3.2.5. Let $T : G \to GL(M, F)$ be an isotypic linear representation of a group G.

(a) T is irreducible if and only if $\operatorname{com}(T)$ is a division algebra.

(b) If the simple algebra $\operatorname{com}(T)$ has a set of primitive idempotents $e_1, e_2, \ldots, e_k$ (with $e_1 + e_2 + \cdots + e_k = 1$ and $e_i e_j = \delta_{i\,j} e_i$ for $1 \leq i, j \leq k$), then T has a set of irreducible components $T_1, T_2, \ldots, T_k$, for $T_i = T|_{M_i} : G \to GL(M_i, F)$ and for $M_i = e_i(M)$, for $1 \leq i \leq k$.

Propositions 3.2.3, 3.2.4(a), and 3.2.5(a) imply the analogous results for a matrix representation $\hat{T} : G \to GL(M, F)$. Thus, $\hat{T}$ is completely reducible if and only if $\operatorname{env}(\hat{T})$ is semi-simple, and $\hat{T}$ completely reducible implies that $\operatorname{com}(\hat{T})$ is semi-simple, that $\hat{T}$ is isotypic if and only if $\operatorname{env}(\hat{T})$ is simple, and that $\hat{T}$ is isotypic if and only if $\operatorname{com}(\hat{T})$ is simple. Finally, if $\hat{T}$ is isotypic, then $\hat{T}$ is irreducible if and only if $\operatorname{com}(\hat{T})$ is a division algebra.

Propositions 3.2.3–3.2.5 follow directly from well known results concerning modules of associative algebras. Further information can be found (for example) in Section 25 of Curtis and Reiner [31].

We next consider the complexity of the problem of obtaining a basis and set of structure constants for the enveloping algebra, or for the commutator algebra, from a description of a matrix representation of a finitely generated group.

Lemma 3.2.6. Let $\hat{T} : G \to GL(m, F)$ be a matrix representation of a group G with generators $g_1, g_2, \ldots, g_n$.

(a) Given the matrices $\hat{T}(g_1), \hat{T}(g_2), \ldots, \hat{T}(g_n)$, a basis and set of structure constants for the matrix algebra $\operatorname{env}(\hat{T})$ can be computed using a polynomial number of field operations, or using arithmetic-Boolean circuits over F of polynomial size and of depth $O(\log^3(nm))$.

(b) Given the matrices $\hat{T}(g_1), \hat{T}(g_2), \ldots, \hat{T}(g_n)$, a basis and set of structure constants for the matrix algebra $\operatorname{com}(\hat{T})$ can be computed using a polynomial number of field operations, or using arithmetic-Boolean circuits over F of polynomial size and of depth $O(\log^2(nm))$.

Proof. Part (a) is an immediate consequence of Theorem 2.2.10; for the algebra $\operatorname{env}(\hat{T})$ is generated by the matrices $\hat{T}(g_1), \hat{T}(g_2), \ldots, \hat{T}(g_n)$. A basis for the algebra $\operatorname{com}(\hat{T})$ is obtained by computing a basis for the set of solutions of a system of homogeneous linear equations (namely, the equations $\hat{T}(g_i)X = X\hat{T}(g_i)$, with the entries of X as indeterminates, for $1 \leq i \leq n$). Hence, by the results of Section 1, a basis for this algebra can be computed at the cost stated above. Structure constants for the algebra can then be computed by solving a further system of linear equations — again, at the stated cost. ■

We will also need to generate a description of the isotypic (respectively, irreducible) components of a matrix representation $\hat{T}$ from a set of central primitive (or, respectively, primitive) idempotents of $\mathrm{env}(\hat{T})$ (respectively, of $\mathrm{com}(\hat{T})$). We next show that these computations can be performed efficiently.

Lemma 3.2.7. *Let $\hat{T} : G \to GL(m, F)$ be a matrix representation of a finitely generated group G.*

(a) *If $\hat{T}$ is completely reducible then, given a set of central primitive idempotents for $\mathrm{env}(\hat{T})$, a set of isotypic components $\hat{T}_1, \hat{T}_2, \ldots, \hat{T}_k$ and a matrix $X \in GL(m, F)$ with $\hat{T}(g) = X^{-1}\mathrm{Diag}(\hat{T}_1(g), \hat{T}_2(g), \ldots, \hat{T}_k(g))X$ for all $g \in G$ can be computed, using arithmetic-Boolean circuits of size $m^{O(1)}$ and of depth $O(\log^2 m)$.*

(b) *If $\hat{T}$ is isotypic then, given a set of primitive idempotents for $\mathrm{com}(\hat{T})$, a set of irreducible components $\hat{T}_1, \hat{T}_2, \ldots, \hat{T}_k$ and a matrix $X \in GL(m, F)$ with $\hat{T}(g) = X^{-1}\mathrm{Diag}(\hat{T}_1(g), \hat{T}_2(g), \ldots, \hat{T}_k(g))X$ for all $g \in G$ can be computed, using arithmetic-Boolean circuits of size $m^{O(1)}$ and of depth $O(\log^2 m)$.*

Proof. We first consider (a). Suppose now that $\hat{T}$ is completely reducible and that we are given a set of central primitive idempotents $e_1, e_2, \ldots, e_k$ for $\mathrm{env}(\hat{T})$. For convenience, we will fix $M = M_{m\times 1}(F)$ as an F-vector space, with basis

$$\epsilon_1 = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}, \quad \epsilon_2 = \begin{bmatrix} 0 \\ 1 \\ \vdots \\ 0 \\ 0 \end{bmatrix}, \quad \cdots, \quad \epsilon_{m-1} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \\ 0 \end{bmatrix}, \quad \epsilon_m = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}.$$

We obtain a linear representation $T : G \to GL(M, F)$ from $\hat{T}$ by choosing as $T(g)$ the linear representation whose coefficient matrix (with respect to the basis $\epsilon_1, \epsilon_2, \ldots, \epsilon_m$) is $\hat{T}(g)$, for all $g \in G$. Now T is clearly a completely reducible linear transformation. Applying Proposition 3.2.4(a), we find that T has isotypic components $T_1, T_2, \ldots, T_k$, where $T_i = T|_{M_i}$, for $M_i = \hat{e}_i(M)$ and for $\hat{e}_i$ the linear transformation whose coefficient matrix with respect to the basis $\epsilon_1, \epsilon_2, \ldots, \epsilon_m$ is e_i.

We obtain a basis for the carrier space M_i by choosing a maximal linearly independent subset of the set of vectors

$$e_i\epsilon_1, \quad e_i\epsilon_2, \quad \cdots \quad e_i\epsilon_m.$$

This operation can be performed sequentially in polynomial time, or in parallel using arithmetic-Boolean circuits of the size and depth stated in the lemma (see

Section 1.3, and note the reductions given in [14]). We set m_i to be the number of vectors in this basis, and denote these vectors by

$$x_{i\,1},\, x_{i\,2},\, \ldots,\, x_{i\,m_i} \in M_{m\times 1}(F).$$

Now since $e_1 + e_2 + \cdots + e_k = 1$ in $\mathrm{env}(\hat{T})$, and $M = M_1 \oplus M_2 \oplus \cdots \oplus M_k$, the column vectors

$$x_{1\,1},\, x_{1\,2},\, \ldots,\, x_{1\,m_1},\, x_{2\,1},\, x_{2\,2},\, \ldots,\, x_{k-1\,m_{k-1}},\, x_{k\,1},\, x_{k\,2},\, \ldots,\, x_{k\,m_k}$$

form a basis (of size m) for M over F. We define X to be the matrix in $M_{m\times m}(F)$ whose i^{th} column is the i^{th} element of this basis, for $1 \le i \le m$. Clearly X is nonsingular. Since each subspace M_i of M is a G-subspace, it is also clear that the matrix

$$XT(g)X^{-1}$$

is block diagonal, with k blocks of sizes $m_1, m_2, \ldots, m_k$, and that the i^{th} block specifies the action of g on the elements of M_i. That is,

$$\hat{T}(g) = X^{-1}\mathrm{Diag}(\hat{T}_1(g),\, \hat{T}_2(g),\, \ldots,\, \hat{T}_k(g))X$$

for all $g \in G$, where the matrix representations $\hat{T}_1, \hat{T}_2, \ldots, \hat{T}_k$ are the isotypic components of $\hat{T}$. The matrix X, and the matrices $\tau_{i\,j} = \hat{T}_i(g_j)$ (for a set $g_1, g_2, \ldots, g_n$ of generators of G) can all be computed at the cost stated in the lemma, as required to prove (a).

The proof of (b) is similar: We apply Proposition 3.2.5(c) instead of Proposition 3.2.4(c), and argue as above. ■

Theorem 3.2.8.

(a) "Identification of a Completely Reducible Representation"
$\preceq_3$ "Isolation of the Radical";
(b) "Isotypic Components of a Completely Reducible Representation"
$\preceq_3$ "Extraction of Simple Components";
(c) "Irreducible Components of an Isotypic Representation"
$\preceq_3$ "Decomposition of a Simple Algebra";

... where we denote by "Problem 1 $\preceq_k$ Problem 2" the fact that "Problem 1" can be solved by solving an instance of "Problem 2" whose size is polynomial in the size of the original instance of "Problem 1", and that the initial conversion of the instance of Problem 1 into an instance of Problem 2, and the computation of the solution of the original instance from the solution of the derived instance, can be performed using arithmetic-Boolean circuits of size $N^{O(1)}$ and depth $O(\log^k N)$, for N the size of the original problem.

Proof. Suppose we are given the images (under a matrix representation $\hat{T}$ of degree m over F) of a set of generators g_1, g_2, ..., g_n for a group G — that is, the input for any of the computational problems defined at the beginning of Section 3.2. Then, by Lemma 3.2.6, we can compute a basis and set of structure constants for each of the algebras $\text{env}(\hat{T})$ and $\text{com}(\hat{T})$ using arithmetic-Boolean circuits of depth $O(\log^3(nm))$ and size polynomial in nm.

Part (a) of the theorem now follows immediately from Proposition 3.2.3(a) (or, to be more precise, from the analogous result for matrix representations). We prove part (b) by applying Proposition 3.2.4(c) and Lemma 3.2.7(a). We prove part (c) by applying Proposition 3.2.5(c) and Lemma 3.2.7(b). ■

We obtain the same "NC^3-reductions" for Boolean computations for matrix representations of finitely generated groups over finite fields and number fields, provided that we represent field elements as discussed in Section 1.2, so that arithmetic (including division of nonzero field elements) can be performed using arithmetic-Boolean circuits of polynomial size and polylogarithmic depth.

Applying these reductions, and the results for associative algebras discussed in Section 2, we obtain the following results.

Corollary 3.2.9. *Let G be a finitely generated group with generators $g_1, g_2, \ldots, g_n$, and let $\hat{T} : G \to GL(m, F)$ be a matrix representation of G over a field F. Suppose we are given the matrices $\hat{T}(g_1)$, $\hat{T}(g_2)$, ..., $\hat{T}(g_n)$.*

(a) *If F has characteristic zero, then we can decide whether $\hat{T}$ is completely reducible using arithmetic-Boolean circuits over F of depth $O(\log^3(mn))$ and size polynomial in mn.*

(b) *If $F = \mathbb{F}_p[t]/(f) \cong \mathbb{F}_{p^l}$ and elements of F are represented (in a description of $\hat{T}$) as vectors of elements of $\mathbb{F}_p$, with each $\alpha \in \mathbb{F}_p[t]/(f)$ represented by the coefficients of a polynomial $\hat{\alpha} \in \mathbb{F}_p[t]$ with degree less than l such that $\alpha = (\hat{\alpha} \bmod f)$, then we can decide whether $\hat{T}$ is completely reducible using arithmetic-Boolean circuits over $\mathbb{F}_p$ of depth $O(\log^3(mnl))$ and size polynomial in mnl.*

(c) *If F is a finite field, or a number field, then given a description of $\hat{T}$ (as described above) of size N we can decide whether $\hat{T}$ is completely reducible using time polynomial in N, or using Boolean circuits of size polynomial in N and depth $O(\log^3 N)$.*

Corollary 3.2.9 is a consequence of Theorem 3.2.8(a) (and the analogous reduction for Boolean computations over finite fields and number fields), as well as Theorem 2.3.4 and Theorem 2.3.17.

Corollary 3.2.10. Let G be a finitely generated group with generators $g_1, g_2, \ldots, g_n$ and let $\hat{T} : G \to GL(m, F)$ be a completely reducible matrix representation of G over a field F. Suppose we are given the matrices $\hat{T}(g_1)$, $\hat{T}(g_2)$, $\ldots$, $\hat{T}(g_n)$.

(a) If F is perfect then we can decide whether $\hat{T}$ is isotypic, and generate the isotypic components of $\hat{T}$ (by solving an instance of the problem "Isotypic Components of a Completely Reducible Representation") using arithmetic-Boolean circuits over F, with oracles for factorisation of squarefree polynomials in $F[t]$, of size $(nm)^{O(1)}$.

(b) If F is a number field then we can decide whether $\hat{T}$ is isotypic, and generate the isotypic components of $\hat{T}$, using $N^{O(1)}$ Boolean operations (for input size N).

(c) If $F = \mathbb{F}_{p^l}$ then we can decide whether $\hat{T}$ is isotypic, and generate the isotypic components of $\hat{T}$, using $(nmpl)^{O(1)}$ Boolean operations, or using a probabilistic Boolean algorithm using $(nml \log p)^{O(1)}$ Boolean operations (that is, in polynomial time), which either successfully performs the above computation, or indicates "failure", failing with probability at most 1/2.

(d) If F is a number field then the matrix representation $\hat{T}_{\mathbb{C}} : G \to GL(m, \mathbb{C})$ is also completely reducible. Furthermore, we can compute integers $k > 0$, and m_1, m_2, $\ldots$, $m_k > 0$ (with $m_1 + m_2 + \cdots m_k = m$), finite extensions E_1, E_2, $\ldots$, E_k of F, a nonsingular matrix $X \in M_{m\times m}(\mathbb{C})$ (such that each entry of X lies in one of the number fields E_1, E_2, $\ldots$, E_k), and isotypic representations $\hat{T}_i : G \to GL(m_i, E_i)$ (presented by the set of matrices $\hat{T}_i(g_1)$, $\hat{T}_i(g_2)$, $\ldots$, $\hat{T}_i(g_n)$), for $1 \le i \le k$, such that $(\hat{T}_1)_{\mathbb{C}}$, $(\hat{T}_2)_{\mathbb{C}}$, $\ldots$, $(\hat{T}_k)_{\mathbb{C}}$ are the isotypic components of $\hat{T}_{\mathbb{C}}$, and so that

$$\hat{T}_{\mathbb{C}}(g) = X^{-1} \mathrm{Diag}((\hat{T}_1)_{\mathbb{C}}(g),\, (\hat{T}_2)_{\mathbb{C}}(g),\, \ldots,\, (\hat{T}_k)_{\mathbb{C}}(g))X$$

for all $g \in G$. This computation can be performed using $N^{O(1)}$ Boolean operations, for input size N.

If, in addition, $F \subseteq \mathbb{R}$, then $\hat{T}_{\mathbb{R}} : G \to GL(m, \mathbb{R})$ is also completely reducible, and the isotypic components of $\hat{T}_{\mathbb{R}}$ can be computed (in the form described above for the components of $\hat{T}_{\mathbb{C}}$) using $N^{O(1)}$ Boolean operations, for input size N.

Corollary 3.2.10 is a consequence of Theorem 3.2.8(b), as well as Theorems 2.4.10, 2.4.11, and 2.4.28. We have also noted here that (for F a number field) if $\hat{T}$ is a completely reducible matrix representation of G over F then $\hat{T}_{\mathbb{C}}$ is completely reducible over $\mathbb{C}$ and, if in addition $F \subseteq \mathbb{R}$, $\hat{T}_{\mathbb{R}}$ is completely reducible over $\mathbb{R}$. These facts follow directly from Proposition 3.2.3(a), Theorem 2.3.20(i), and the observation that the algebras $(\text{env}(\hat{T}))_{\mathbb{C}}$ and $(\text{env}(\hat{T}_{\mathbb{C}}))$ are isomorphic over $\mathbb{C}$ (and that the algebras $(\text{env}(\hat{T}))_{\mathbb{R}}$ and $(\text{env}(\hat{T}_{\mathbb{R}}))$ are isomorphic over R, if $F \subseteq \mathbb{R}$).

Corollary 3.2.11. Let G be a finitely generated group with generators $g_1, g_2, \ldots, g_n$ and let $\hat{T} : G \to GL(m, F)$ be an isotypic matrix representation of G over a field F. Suppose we are given the matrices $\hat{T}(g_1)$, $\hat{T}(g_2)$, …, $\hat{T}(g_n)$.

(a) If $F = \mathbb{F}_{p^l}$ then we can decide whether $\hat{T}$ is irreducible, and generate a set of irreducible components of $\hat{T}$ (by solving an instance of the problem "Irreducible Components of an Isotypic Representation"), using a probabilistic Boolean algorithm which either successfully computes the desired components or returns "failure" (with probability at most 1/2), in time $(nml \log p)^{O(1)}$ (that is, in polynomial time).

(b) If F is a number field then the matrix representation $\hat{T}_{\mathbb{C}} : G \to GL(m, \mathbb{C})$ is also isotypic. Furthermore, we can compute integers k and l such that $kl = m$, finite extensions E_1, E_2, …, E_k of F, a nonsingular matrix $X \in M_{m\times M}(\mathbb{C})$ (such that each entry of X lies in one of the number fields E_1, E_2, …, E_k), and irreducible representations $\hat{T}_i : G \to GL(l, E_i)$ (presented by the matrices $\hat{T}_i(g_1)$, $\hat{T}_i(g_2)$, …, $\hat{T}_i(g_n)$), such that $(\hat{T}_1)_{\mathbb{C}}$, $(\hat{T}_2)_{\mathbb{C}}$, …, $(\hat{T}_k)_{\mathbb{C}}$ form a set of irreducible components of $\hat{T}_{\mathbb{C}}$, and

$$\hat{T}_{\mathbb{C}}(g) = X^{-1}\text{Diag}((\hat{T}_1)_{\mathbb{C}}(g),\, (\hat{T}_2)_{\mathbb{C}}(g),\, \ldots,\, (\hat{T}_k)_{\mathbb{C}}(g))X$$

for all $g \in G$. This computation can be performed using a probabilistic Boolean algorithm using $N^{O(1)}$ operations (for input size N), which either returns the desired components or reports "failure" (the latter with probability at most 1/2).

If, in addition, $F \subseteq \mathbb{R}$, then $\hat{T}_{\mathbb{R}} : G \to GL(m, \mathbb{R})$ is also isotypic, and a set of irreducible components of $\hat{T}_{\mathbb{R}}$ can be computed (in the form described above for components of $\hat{T}_{\mathbb{C}}$) using a probabilistic Boolean algorithm using $N^{O(1)}$ operations (for input size N), which either returns the desired components or reports "failure" (the latter with probability at most 1/2).

Corollary 3.2.11 is a consequence of Theorem 3.2.8(c), as well as Theorems 2.5.9, 2.5.11, and 2.5.18.

As we note in Section 2, Babai and Rónyai have improved the results stated in Theorems 2.5.11 and 2.5.8 (see [5]). Their methods can be used to improve on Corollary 3.2.11(b): The field extensions E_1, E_2, …, E_k described here can be replaced by a *single* extension $E \supseteq F$ whose degree over F is polynomial in m. Consequently, all the irreducible components of $\hat{T}_{\mathbb{C}}$ (or of $\hat{T}_{\mathbb{R}}$, if $F \subseteq \mathbb{R}$) can be presented as matrix representations over a *single* extension E, in probabilistic polynomial time. (Note in particular Theorem 1.3 of [5].)

We would like to have reductions in the other direction as well (in particular, for the third of these three sets of problems: *cf.* Propositions 2.5.21 and 2.5.22). We will provide these reductions for infinite fields.

Henceforth, we assume that the field F is infinite. We first note that any finite dimensional associative algebra $A \subseteq M_{m\times m}(F)$ of dimension n over F can be expressed as the enveloping algebra of a matrix representation $\hat{T} : G \to GL(m, F)$, where G is a free group with n generators $g_1, g_2, \ldots, g_n$.

Lemma 3.2.12. Let $A \subseteq M_{m\times m}(F)$ be an associative algebra over an infinite field F with dimension n and basis $a_1, a_2, \ldots, a_n$ over F. Then there exist matrices $\tau_1, \tau_2, \ldots, \tau_n \in M_{m\times m}(F)$ such that $\tau_i = \hat{T}(g_i)$, for a matrix representation $\hat{T} : G \to GL(m, F)$ of the free group G with generators $g_1, g_2, \ldots, g_n$, such that $A = \text{env}(\hat{T})$. The matrices $\tau_1, \tau_2, \ldots, \tau_n$ can be computed from the basis $a_1, a_2, \ldots, a_n$ using arithmetic-Boolean circuits over F of depth $O(\log^2(nm))$ and size polynomial in nm.

Proof. Suppose first that the matrices $a_1, a_2, \ldots, a_n$ are all nonsingular. Then it is sufficient to set $\tau_i = \hat{T}(g_i) = a_i$ for $1 \le i \le n$. Clearly, $\hat{T} : G \to GL(m, F)$ is a matrix representation of the free group G. Since each $a_i = \hat{T}(g_i) \in \text{env}(\hat{T})$, and $a_1, a_2, \ldots, a_n$ is a basis for A over F, $A \subseteq \text{env}(\hat{T})$. Conversely, if $g \in G$ then

$$g = h_1 h_2 \cdots h_r$$

for some integer $r > 0$, such that $g_i \in \{ g_1, g_2, \ldots, g_n, g_1^{-1}, g_2^{-1}, \ldots, g_n^{-1} \}$ (note that $1_G = g_1 g_1^{-1}$). Clearly, since $\hat{T}(g_i) = a_i \in A$ and $\hat{T}(g_i^{-1}) = a_i^{-1} \in F[a_i] \subseteq A$, for $1 \le i \le n$, $\hat{T}(h_i) \in A$ for $1 \le i \le r$, and $\hat{T}(g) \in A$. Thus $\text{env}(\hat{T}) \subseteq A$ (since A is an F-subalgebra of $M_{m\times m}(F)$). Then $A = \text{env}(\hat{T})$ as desired, and the matrices $\tau_1, \tau_2, \ldots, \tau_n$ can clearly be computed (in this case) at the desired cost. It is also clear that we can find a matrix representation $\hat{T}$ with the desired properties from an *arbitrary* basis for A, if this basis can be used to generate a second basis for A which consists of nonsingular matrices.

We now consider the general case. Given an arbitrary basis $a_1, a_2, \ldots, a_n$, we obtain a second basis $b_1, b_2, \ldots, b_n$ for A such that b_1 is the identity matrix in $M_{m\times m}(F)$, by setting b_1 to be this matrix, setting b_{i+1} to be a_i for all i less than n such that the matrices $b_1, a_1, a_2, \ldots, a_i$ are linearly independent, and by setting $b_{i+1} = a_{i+1}$ for all other i. Then the new basis has the form

$$1_A, a_1, a_2, \ldots, a_{i-2}, a_{i-1}, a_{i+1}, a_{i+2}, \ldots, a_n$$

for some integer i such that a_i is an F-linear combination of the first i matrices in this basis — so that $b_1, b_2, \ldots, b_n$ is a basis for A.

Note that the integer i can be computed in polynomial time, or in parallel at the cost stated in the lemma, by solving n (singular) systems of linear equations in parallel (see Section 1.3 for more information).

We now obtain a third basis $c_1, c_2, \ldots, c_n$ for A over F, consisting of nonsingular matrices, by setting $c_1 = b_1 = 1_A$ and by setting $c_i = b_i + \lambda_i 1_A$, for $\lambda_i \in F$ such that $\det(b_i + \lambda_i 1_A) \neq 0$, for $2 \leq i \leq n$. Since $A \subseteq M_{m\times m}(F)$, it is clear that there are at most m elements γ of F such that $b_i + \gamma 1_A$ is singular. Hence we can find an appropriate field element λ_i, and the basis element c_i, by considering any set of $m+1$ elements of F as candidates for λ_i. Since F is infinite, some such set of elements exists. The condition "$\det(b_i + \gamma 1_A) = 0$?" can be tested at the cost stated in the lemma, and the basis elements $c_2, c_3, \ldots, c_n$ can be generated independently (in parallel). Thus the basis $c_1, c_2, \ldots, c_n$ can be computed at the cost stated in the lemma — as required to complete the proof. ■

We will also need to represent an arbitrary finite dimensional semi-simple associative matrix algebra A as $\mathrm{com}(\hat{T})$, for some completely reducible matrix representation $\hat{T}$ of a finitely generated free group. To show that this is possible, we use a concept (for algebras) similar to that of the commutator algebra of a representation.

Definition 3.2.13. Let $A \subseteq M_{m\times m}(F)$ be an associative matrix algebra. The *centralizer* of A, $\mathrm{cent}(A)$, is the algebra of matrices in $M_{m\times m}(F)$ which commute with every element of A. That is, $b \in \mathrm{cent}(A)$ if and only if $ab = ba$ for all $a \in A$, for $b \in M_{m\times m}(F)$. A is said to have the *double centralizer property* if $\mathrm{cent}(\mathrm{cent}(A)) = A$.

Note that if $\hat{T} : G \to GL(m, F)$ is a matrix representation of a group G, then $\mathrm{com}(\hat{T}) = \mathrm{cent}(\mathrm{env}(\hat{T}))$.

Example 3.2.14. Consider the algebra $A \subseteq M_{2\times 2}(F)$ of 2×2 upper triangular matrices, with basis

$$a_1 = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, \quad a_2 = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad a_3 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$$

over F. Let α, β, γ, $\delta \in F$; then

$$x = \begin{bmatrix} \alpha & \beta \\ \gamma & \delta \end{bmatrix} \in \mathrm{cent}(A)$$

if and only if $xa_i = a_i x$ for $1 \leq i \leq 3$. It is easily checked that this is the case if and only if $\beta = \gamma = 0$ and $\alpha = \delta$, so that x is a scalar multiple of the identity matrix. Thus

$$\mathrm{cent}(A) = \{\, \alpha I_2 \;:\; \alpha \in F \,\}, \quad \text{and} \quad \mathrm{cent}(\mathrm{cent}(A)) = M_{2\times 2}(F) \neq A.$$

Thus A *does not* have the double centralizer property.

Note also that we have already shown that there exists a matrix representation $\hat{T} : G \to GL(2, F)$, for G the free group with 3 generators, such that $\text{env}(\hat{T}) = A$. Clearly $\hat{T}$ is not completely reducible (by Proposition 3.2.3), but $\text{com}(\hat{T}) \cong F$ is semi-simple — so, in general, $\text{com}(\hat{T})$ semi-simple does not imply that $\hat{T}$ is completely reducible.

Proposition 3.2.15. Let $A \subseteq M_{m\times m}(F)$ be a semi-simple associative algebra over F; then A has the double centralizer property.

For a proof of this see, for example, Jacobson [68] (Theorem 4.10).

We use this proposition to prove the following lemma.

Lemma 3.2.16. Let $A \subseteq M_{m\times m}(F)$ be a semi-simple associative algebra over F, with dimension n and basis $a_1, a_2, \ldots, a_n$ over F. Then there exists an integer k with $0 \le k \le m^2$, a free group G with generators $g_1, g_2, \ldots, g_k$, and a completely reducible matrix representation $\hat{T} : G \to GL(m, F)$, such that $A = \text{com}(\hat{T})$. Furthermore, the integer k and matrices $\hat{T}(g_1), \hat{T}(g_2), \ldots, \hat{T}(g_k)$ can be computed from the basis $a_1, a_2, \ldots, a_n$ for A using arithmetic-Boolean circuits of size polynomial in m and depth $O(\log^2 m)$.

Proof. Let A be as stated in the lemma; then Lemma 3.2.12 implies that $A = \text{env}(\hat{U})$ for some matrix representation $\hat{U}$ of a finitely generated (free) group G_1 over F. By Proposition 3.2.3(a), the representation $\hat{U}$ is completely reducible — and hence $\text{com}(\hat{T}) = \text{cent}(\text{env}(\hat{U})) = \text{cent}(A)$ is semi-simple, by Proposition 3.2.3(b).

Let k be the dimension of $\text{cent}(A)$ over F, and let $b_1, b_2, \ldots, b_k$ be a basis for $\text{cent}(A)$ over F. These can be computed from the basis $a_1, a_2, \ldots, a_n$ for A by solving systems of linear equations over F, using arithmetic-Boolean circuits of size polynomial in m and depth $O(\log^2 m)$. Solving further linear systems, we compute a set of structure constants for this basis, also at this cost. We now apply Lemma 3.2.12 again, to conclude that there exists a finite-dimensional matrix representation $\hat{T} : G \to GL(m, F)$, for a free group G with generators $g_1, g_2, \ldots, g_k$, such that $\text{env}(\hat{T}) = \text{cent}(A)$. Now $\hat{T}$ is completely reducible (by Proposition 3.2.3(a)), and $\text{com}(\hat{T}) = \text{cent}(\text{env}(\hat{T})) = \text{cent}(\text{cent}(A)) = A$ (by Proposition 3.2.15, since A is semi-simple), as required to complete the proof. ■

We are now ready to state and prove our second set of reductions.

Theorem 3.2.17

(i) "Extraction of Simple Components"

$\preceq_2$ "Isotypic Components of a Completely Reducible Representation";

(ii) "Decomposition of a Simple Algebra"

$\preceq_2$ "Irreducible Components of an Isotypic Representation";

... where we use $\preceq_k$ as described in the statement of Theorem 3.2.8, and we consider computations over infinite fields F.

Proof. We first consider (a). Assume now that $A \subseteq M_{m\times m}(F)$ is a semi-simple associative algebra of dimension n over an infinite field F, and that we are given a basis $a_1, a_2, \ldots, a_n$ for A over F. It follows by Lemma 3.2.12 that we can compute a completely reducible matrix representation $\hat{T} : G \to GL(m, F)$ of a finitely generated group G, such that $A = \text{env}(\hat{T})$, using arithmetic-Boolean circuits of size polynomial in mn and depth $O(\log^3(mn))$. Solving an instance of the problem "Isotypic Components of a Completely Reducible Representation" (with input $\hat{T}$), we obtain a matrix $X \in GL(m, F)$ such that for all $g \in G$,

$$\hat{T}(g) = X^{-1}\text{Diag}(\hat{T}_1(g), \hat{T}_2(g), \ldots, \hat{T}_k(g))X,$$

where $\hat{T}_1, \hat{T}_2, \ldots, \hat{T}_k$ are the isotypic components of $\hat{T}$. We also obtain the number of components, k, and the degree m_i of the component $\hat{T}_i$, for $1 \le i \le k$.

For $1 \le i \le k$, set

$$e_i = X^{-1}\text{Diag}(0_{m_1}, 0_{m_2}, \ldots, 0_{m_{i-1}}, 1_{m_i}, 0_{m_{i+1}}, \ldots, 0_{m_k})X,$$

where we denote by 0_s the zero matrix in $M_{s\times s}(F)$, and by 1_s the identity matrix in $GL(s, F)$. It is clear that each idempotent e_i commutes with $\hat{T}(g)$ for all $g \in G$, so that $e_i \in \text{com}(\hat{T})$, and that $e_1, e_2, \ldots, e_k$ are the *only* idempotents in $\text{com}(\hat{T})$ with the properties described as holding for the central primitive idempotents of $\text{env}(\hat{T})$ in Proposition 3.2.4(b). It follows, by that proposition, that $e_1, e_2, \ldots, e_k$ form a set of central primitive idempotents of $\text{env}(\hat{T})$. Clearly, these idempotents can be computed from the matrices X and X^{-1} and the integers k and $m_1, m_2, \ldots, m_k$, using arithmetic-Boolean circuits over F of size polynomial and depth logarithmic in m. By Theorem 2.4.4, bases for the simple components of $A = \text{env}(\hat{T})$ can then be computed using arithmetic-Boolean circuits of size polynomial in m and depth $O(\log^2 m)$, as required to complete the proof of part (a).

The proof of (b) is similar. Suppose now that $A \subseteq M_{m\times m}(F)$ is a simple algebra, and that we are given a basis $a_1, a_2, \ldots, a_n$ for A over F. By Lemma 3.2.16, we can compute a completely reducible matrix representation $\hat{T} : G \to GL(m, F)$, for a finitely generated group G, such that $A = \mathrm{com}(\hat{T})$, using arithmetic-Boolean circuits over F of size polynomial in m and depth $O(\log^2 m)$. It follows by Proposition 3.2.4(b) that $\hat{T}$ is an isotypic representation.

Solving an instance of the problem "Irreducible Components of an Isotypic Representation" (with input $\hat{T}$) we obtain a matrix $X \in GL(m, F)$ such that for all $g \in G$,

$$\hat{T}(g) = X^{-1}\mathrm{Diag}(\hat{T}_1(g), \hat{T}_2(g), \ldots, \hat{T}_k(g))X,$$

where $\hat{T}_1, \hat{T}_2, \ldots, \hat{T}_k$ form a set of irreducible components of $\hat{T}$. We also obtain the number of components k, and the degree l of each of these components.

We now set

$$e_i = X^{-1}\mathrm{Diag}(0_l, 0_l, \ldots, 0_l, 1_l, 0_l, \ldots, 0_l)X,$$

with the identity matrix 1_l in the i^{th} block of the diagonal matrix shown here, and use Proposition 3.2.5(b) (instead of Proposition 3.2.4(c)) to show that $e_1, e_2, \ldots, e_l$ is a set of primitive idempotents in $\mathrm{com}(\hat{T})$. (The argument is similar to the one given in the proof of part (a).) Finally, we use Theorem 2.5.3 to complete the proof of (b). ■

Again, the reductions are also correct for Boolean computations over number fields (and large finite fields).

Corollary 3.2.18.

(a) Assuming the Generalised Riemann Hypothesis (GRH), there exists a Las Vegas polynomial time reduction from the problem of deciding whether a positive integer m is a quadratic residue modulo a squarefree integer n, to deciding whether an isotypic matrix representation $\hat{T} : G \to GL(4, \mathbb{Q})$ of a finitely generated group G over $\mathbb{Q}$ is an irreducible representation.

(b) Assuming GRH, there exists a randomised polynomial time reduction from the problem of factoring squarefree integers to the problem of computing a set of irreducible components of an isotypic matrix representation $\hat{T} : G \to GL(4, \mathbb{Q})$ of a finitely generated group G over $\mathbb{Q}$.

These facts follow from Theorem 3.2.17 and the results of Rónyai ([103], [104]), stated here as Propositions 2.5.21 and 2.5.22.

The computational problems of decomposing matrix representations of finitely generated groups and finite groups over a field F have previously been studied for the cases $F = \mathbb{R}$ and $F = \mathbb{C}$. In particular, Gabriel developed methods for the decomposition of a completely reducible matrix representation $\hat{T} : G \to GL(m, \mathbb{R})$ or $\hat{T} : G \to GL(m, \mathbb{C})$, given as input matrices $\hat{T}(g_1), \hat{T}(g_2), \ldots, \hat{T}(g_n)$, where $g_1, g_2, \ldots, g_n$ is a set of generators of G (see [45]–[49]). Like the methods discussed in this section, Gabriel's methods are correct only if the input is exact (rather than a numerical estimate), and if exact arithmetic is used. Hence we cannot implement Gabriel's method correctly, using exact computations, without dealing with the problems of representing elements of $\mathbb{R}$ and $\mathbb{C}$ discussed in Section 1. Further, since Gabriel's methods are iterative, it is not clear that we could obtain asymptotically fast algorithms from his methods by representing real and complex numbers as described in that section. Note, however, that Gabriel *does* use the commutator algebra of a completely reducible matrix representation to decompose the representation in the manner described in this section (in particular, as indicated by Proposition 3.2.5 and in the proofs of Theorems 3.2.8 and 3.2.17). To some extent, the reductions given in this section are extensions (and adaptations) of reductions which are implicit in Gabriel's work.

Dixon considers related problems involving matrix representations of finite groups, to which numerical techniques can be applied more successfully (see [35]). We will discuss his work further in Section 3.3.5.

3.2.2. Equivalence of Matrix Representations

To this point we have noted similarities between computational problems for decompositions of finite dimensional associative matrix algebras, and for decompositions of matrix representations of finitely generated groups. We now note a pair of corresponding problems (one for algebras, the other for representations) whose complexities appear to differ.

The problems we consider are those of deciding whether two associative algebras are isomorphic, and of deciding whether two matrix representations of the same group are equivalent (the problem "Equivalence of Representations" defined at the beginning of Section 3.2). We first consider the latter problem. For an infinite field F, and a finitely generated group G with generators g_1, g_2, …, g_n, we are given as inputs the matrices

$$\hat{T}_1(g_1),\, \hat{T}_1(g_2),\, \ldots,\, \hat{T}_1(g_n), \qquad \text{and} \qquad \hat{T}_2(g_1),\, \hat{T}_2(g_2),\, \ldots,\, \hat{T}_2(g_n),$$

for matrix representations $\hat{T}_1$, $\hat{T}_2 : G \to GL(m, F)$. Now $\hat{T}_1$ and $\hat{T}_2$ are equivalent if and only if there exists a nonsingular matrix $S \in GL(m, F)$ such that

$$\hat{T}_1(g_i)S = S\hat{T}_2(g_i) \qquad \text{for } 1 \le i \le n.$$

If we consider a matrix $X = (x_{i\,j})_{1 \le i,\, j \le n}$, whose entries are indeterminates, we see that a basis for the space of solutions of the homogeneous system of linear equations (in the $x_{i\,j}$'s)

$$\hat{T}_1(g_i)X = X\hat{T}_2(g_i) \qquad \text{for } 1 \le i \le n$$

can be computed using arithmetic-Boolean circuits of size $(nm)^{O(1)}$ and of depth $O(\log^2(nm))$. Suppose the basis (over F)

$$S_1,\, S_2,\, \ldots,\, S_k$$

of matrices in $M_{m\times m}(F)$ is generated. Then $\hat{T}_1$ and $\hat{T}_2$ are equivalent if and only if there exist elements $\lambda_1,\, \lambda_2,\, \ldots,\, \lambda_k$ of F such that

$$\lambda_1 S_1 + \lambda_2 S_2 + \cdots + \lambda_k S_k$$

is nonsingular; the original problem reduces to the problem of deciding whether such a set of λ_i's exists.

We will not give the details of the solution of this problem, since it is discussed elsewhere. Kaltofen, Krishnamoorthy, and Saunders [72] solve the above problem, using the lemma of Schwartz [111] (Proposition 2.4.23), and thereby obtain a fast parallel algorithm for deciding whether two matrices are similar.* We note the results for our problem below.

Theorem 3.2.19. Let F and G be as above, and suppose we are given matrices $\hat{T}_1(g_i)$ and $\hat{T}_2(g_i)$ for $1 \leq i \leq n$, for generators $g_1,\, g_2,\, \ldots,\, g_n$ of G. Suppose also that we are given an error tolerance ϵ, with $1 > \epsilon > 0$. Then we can decide whether $\hat{T}_1$ and $\hat{T}_2$ are equivalent using a family of probabilistic arithmetic-Boolean circuits over F of size polynomial, and depth polylogarithmic, in nm (with "oracle" nodes in the circuit choosing elements randomly from a finite subset of F of size polynomial in $\lceil nm\epsilon^{-1} \rceil$) and with probability of error less than ϵ.

If F is a number field then the computation can be performed using probabilistic Boolean circuits of size polynomial and depth polylogarithmic in the input size (and, again, with arbitrarily small probability of error).

In contrast, the problem of deciding whether matrix algebras are isomorphic has as a special case the problem of deciding whether a simple algebra of dimension 4 over $\mathbb{Q}$ is isomorphic to $M_{2\times 2}(\mathbb{Q})$. As we noted in Corollary 3.2.18, Rónyai's results can be applied with our reductions of Section 3.2.1 to show that this problem is as hard as the problem of deciding quadratic residuosity for squarefree integers (using probabilistic polynomial-time Boolean computations, assuming the Extended Riemann Hypothesis). Apparently, the extra information given for matrix representations (associating each element of g to a pair $\hat{T}_1(g)$ and $\hat{T}_2(g)$, and requiring

* In fact, this is used to prove that the matrices are similar, or to conclude that they are probably not singular; Kaltofen, Krishnamoorthy, and Saunders use different means, which do not apply to our problem, to prove that matrices are not singular. Their algorithm gives a correct answer with arbitrarily high probability, or detects failure.

that $\hat{T}_1(g)$ be mapped to $\hat{T}_2(g)$, rather than to an arbitrary element of $\mathrm{env}(\hat{T}_2)$), and the differences in the definitions of isomorphism (equivalence) for matrix algebras and for matrix representations, are sufficient to cause the complexity of the problems to differ.

3.3. Matrix Representations and Characters of Finite Groups

In this section we consider matrix representations and characters of (small) finite groups: We assume (initially) that we are given a multiplication table of a group G as part of our input — so that the input size of our problems is at least n^2, for n the number of elements in the group.

We will consider both the problems involving matrix representations introduced (for finitely generated groups) in Section 3.2, and the computation of a character table for a finite group. The main result of this section is that a character table of a finite group can be computed efficiently both sequentially (in polynomial time), and in parallel — in particular, using Boolean circuits of size polynomial, and depth polylogarithmic, in the size of a multiplication table for the input group (see in particular Theorems 3.3.24 and 3.3.27). We obtain these by providing (new) analyses of existing algorithms. Using these facts, we show that the isotypic components of a matrix representation of a finite group can also be computed at this cost (again, if we are given a multiplication table for the group; see Theorem 3.3.31).

We begin with a section of standard material, leading to the definition of a character table for a group; a reader who is familiar with the definitions of character and character table can safely skip Section 3.3.1. We continue, in Section 3.3.2, by presenting and analysing Burnside's algorithm; we note that this is a polynomial time algorithm for the computation of character tables (see Theorem 3.3.24). Modifications of Burnside's algorithm are discussed in Sections 3.3.3 and 3.3.4. We discuss Dixon's algorithm, and show that it has an efficient parallel implementation, in Section 3.3.3. In Section 3.3.4, we present a new modification of this method. While our new algorithm may actually be slightly *less* efficient than Dixon's, it is possible to prove that ours is efficient without recourse to any unproved number theoretic hypotheses. We apply these results to problems for matrix representations of finite groups in Section 3.3.5.

3.3.1. Basic Definitions

We begin with definitions and examples of characters and character tables. The material discussed here is standard; for other treatments, see (for example) [31] or [112].

Definition 3.3.1. Let $T : G \to GL(n, F)$ be a matrix representation of a group G; the *character* of the representation T is the function $\chi_T : G \to F$ such that

$$\chi_T(g) = \mathrm{Trace}(T(g)) = \sum_{i=1}^{n} T(g)_{i\,i} \qquad \text{for } g \in G,$$

where Trace(X) denotes the sum of the diagonal entries of the matrix $X = (X_{i\,j})_{1 \le i,\, j \le n}$.

Example 3.3.2. Consider the group $G = D_3$ and the matrix representation $\hat{T} : G \to GL(6, \mathbb{C})$ of Example 3.1.20. The character $\chi_{\hat{T}}$ of this representation is as follows.

$$\begin{array}{lll} \chi_{\hat{T}}(1) = 6; & \chi_{\hat{T}}(a) = 0; & \chi_{\hat{T}}(a^2) = 0; \\ \chi_{\hat{T}}(b) = 0; & \chi_{\hat{T}}(ab) = 0; & \chi_{\hat{T}}(a^2b) = 0. \end{array}$$

We showed in that example that $\hat{T} = T_1 \oplus T_2 \oplus T_{3\,1} \oplus T_{3\,2}$ for irreducible representations T_1, $T_2 : G \to GL(1, \mathbb{C})$ and for equivalent irreducible representations $T_{3\,1}$, $T_{3\,2} : G \to GL(2, \mathbb{C})$. The characters $\chi_1 = \chi_{T_1}$, $\chi_2 = \chi_{T_2}$, and $\chi_3 = \chi_{T_{3\,1}} = \chi_{T_{3\,2}}$ of these representations are as follows.

g:	1	a	a^2	b	ab	a^2b
$\chi_1(g)$:	1	1	1	1	1	1
$\chi_2(g)$:	1	1	1	-1	-1	-1
$\chi_3(g)$:	2	-1	-1	0	0	0

These are obtained by computing the trace of each matrix $T_i(g)$ (with $T_3(g) = T_{3\,1}(g) = T_{3\,2}(g)$) and using the identity $1 + \omega + \omega^2 = 0$, for ω a primitive 3rd root of unity.

Note that characters are not generally multiplicative over G. In particular, in the above example, $1 = (\chi_3(a))^2 \neq \chi_3(a^2) = -1$.

We now note some useful facts about characters of representations of (arbitrary) groups.

Let $\hat{T}_1$ and $\hat{T}_2$ be two matrix representations of degree m for a group G over a field F, and suppose $\hat{T}_1$ and $\hat{T}_2$ are equivalent. Then there exists a nonsingular matrix $X \in GL(m, F)$ such that

$$\hat{T}_1(g) = X^{-1} \cdot \hat{T}_2(g) \cdot X \qquad \text{for all } g \in G.$$

It is easily checked (using the definition of the trace of a matrix as the sum of its diagonal entries) that if A, $B \in M_{m\times m}(F)$ then $\mathrm{Trace}(AB) = \mathrm{Trace}(BA)$. It follows that

$$\begin{aligned} \chi_{\hat{T}_1}(g) &= \mathrm{Trace}(X^{-1} \cdot \hat{T}_2(g) \cdot X) \\ &= \mathrm{Trace}(X^{-1} \cdot X \cdot \hat{T}_2(g)) \\ &= \mathrm{Trace}(\hat{T}_2(g)) = \chi_{\hat{T}_2}(g). \end{aligned}$$

We have proved the following fact.

Proposition 3.3.3. The characters of equivalent matrix representations of a group G over a field F are equal.

This allows us to define the character of a linear representation in terms of characters of matrix representations.

Definition 3.3.4. Let $T : G \to GL(M, F)$ be a linear representation of a group G, with M an F-vector space of dimension m over F. The *character* of the linear representation T is the character $\chi_{\hat{T}}$ of any matrix representation $\hat{T} : G \to GL(m, F)$ for G over F obtained from T, with respect to a basis for M over F.

Consider again a matrix representation $\hat{T} : G \to GL(m, F)$. Suppose g_1 and g_2 are conjugates in G, so that there exists an element x of G with $g_2 = x^{-1}gx$. Then

$$\begin{aligned}
\chi_{\hat{T}}(g_2) &= \chi_{\hat{T}}(x^{-1}g_1x) \\
&= \mathrm{Trace}(\hat{T}(x^{-1}g_1x)) \\
&= \mathrm{Trace}(\hat{T}(x^{-1}) \cdot \hat{T}(g_1x)) \\
&= \mathrm{Trace}(\hat{T}(g_1x) \cdot T(x^{-1})) \\
&= \mathrm{Trace}(\hat{T}(g_1xx^{-1})) \\
&= \mathrm{Trace}(\hat{T}(g_1)) = \chi_{\hat{T}}(g_1).
\end{aligned}$$

Proposition 3.3.5. The characters of G are *class functions* for G. That is, they induce well defined functions from the set of conjugacy classes of G to the field F.

Example 3.3.6. Consider again the dihedral group D_3. The group has three conjugacy classes, (1), (a), and (b), with

$$\begin{array}{lcll}
(1) & = & \{\,1\,\}, & \\
(a) & = & \{\,a,\, a^2\,\}, & (a^2 = b \cdot a \cdot b^{-1}) \\
(b) & = & \{\,b,\, ab,\, ab^2\,\}. & (ab = (a^2b) \cdot b \cdot (a^2b)^{-1};\ a^2b = (ab) \cdot b \cdot (ab)^{-1})
\end{array}$$

The characters χ_1, χ_2, and χ_3 of G listed in Example 3.3.2 induce functions $\bar{\chi}_1$, $\bar{\chi}_2$, and $\bar{\chi}_3$ respectively, for $\chi_i : C \to F$, $C = \{\,(1),\,(a),\,(b)\,\}$, as follows.

(g):	(1)	(a)	(b)
$\bar{\chi}_1((g))$:	1	1	1
$\bar{\chi}_2((g))$:	1	1	-1
$\bar{\chi}_3((g))$:	2	-1	0

Example 3.3.7. Let $\hat{T}_1 : G \to GL(m_1, F)$ and $\hat{T}_2 : G \to GL(m_2, F)$ be matrix representations, with respective characters $\chi_{\hat{T}_1}, \chi_{\hat{T}_2} : G \to F$. We have discussed matrix representations

$$(T_1 \oplus T_2) : G \to GL(m_1 + m_2, F) \quad \text{and} \quad (T_1 \otimes T_2) : G \to GL(m_1 \cdot m_2, F)$$

in Section 3.1 (see, in particular, Examples 3.1.7 and 3.1.8). The matrices $(T_1 \oplus T_2)(g)$ and $(T_1 \otimes T_2)(g)$ are as follows, for $g \in G$.

$$(T_1 \oplus T_2)(g) = \begin{bmatrix} T_1(g) & 0 \\ 0 & T_2(g) \end{bmatrix},$$

and

$$(T_1 \otimes T_2)(g) = \begin{bmatrix} \beta_{1\,1}T_1(g) & \beta_{1\,2}T_1(g) & \cdots & \beta_{1\,m_2}T_1(g) \\ \beta_{2\,1}T_1(g) & \beta_{2\,2}T_1(g) & \cdots & \beta_{2\,m_2}T_1(g) \\ \vdots & \vdots & \ddots & \vdots \\ \beta_{m_2\,1}T_1(g) & \beta_{m_2\,2}T_1(g) & \cdots & \beta_{m_2\,m_2}T_1(g) \end{bmatrix},$$

such that $\beta_{i\,j} = T_2(g)_{i\,j}$ for $1 \leq i,\, j \leq m_2$. It is evident that

$$\begin{aligned} \chi_{(T_1 \oplus T_2)}(g) &= \text{Trace}((T_1 \oplus T_2)(g)) \\ &= \text{Trace}(T_1(g)) + \text{Trace}(T_2(g)) \\ &= \chi_{T_1}(g) + \chi_{T_2}(g), \end{aligned}$$

and

$$\begin{aligned} \chi_{(T_1 \otimes T_2)}(g) &= \text{Trace}((T_1 \otimes T_2)(g)) \\ &= \sum_{i=1}^{m_2} (\beta_{i\,i} \cdot \text{Trace}(T_1(g))) \\ &= (\text{Trace}(T_1(g))) \cdot \sum_{i=1}^{m_2} (\beta_{i\,i}) \\ &= (\text{Trace}(T_1(g))) \cdot (\text{Trace}(T_2(g))) \\ &= \chi_{T_1}(g) \times \chi_{T_2}(g). \end{aligned}$$

Thus the characters of the direct sum and tensor product of the representations T_1 and T_2 are, respectively, the sum and product of the characters of T_1 and T_2.

We carry our notation for representations over to characters.

Definition 3.3.8. A character $\chi : G \to F$ of a group G is an *irreducible character* (respectively, an *isotypic character,* or a *completely reducible character*) if it is the character χ_T corresponding to some irreducible (respectively, isotypic, or completely reducible) matrix representation T of G over F.

Clearly every character of an irreducible (respectively, isotypic, or completely reducible) matrix representation is, by definition, itself irreducible (respectively, isotypic, or completely reducible). We note that the converse is not generally true, since a character χ can correspond to several inequivalent representations.

Example 3.3.9. Suppose $G = \{ x^i \ : \ i \in \mathbb{N} \}$ is an infinite cyclic group with generator x, and consider the function $\chi : G \to \mathbb{Q}$ such that $\chi(x^i) = 2$ for all $i \in \mathbb{N}$. Then $\chi = \chi_{T_1} = \chi_{T_2}$, for matrix representations T_1, $T_2 : G \to GL(2, \mathbb{Q})$, such that, for all $i \in \mathbb{N}$,

$$T_1(x^i) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \qquad \text{and} \qquad T_2(x^i) = \begin{bmatrix} 1 & i \\ 0 & 1 \end{bmatrix}.$$

Clearly T_1 is completely reducible, and isotypic; hence so is $\chi = \chi_{T_1} = \chi_{T_2}$. However, T_2 is neither completely reducible nor isotypic.

Our remarks about the characters of direct sums of two representations generalise to direct sums of an arbitrary (finite) number of representations. That is, suppose

$$T = T_1 \oplus T_2 \oplus \cdots \oplus T_k$$

for matrix representations $T : G \to GL(m, F)$ and $T_i : G \to GL(m_i, F)$, for $k \geq 0$ and $1 \leq i \leq k$, with $m = m_1 + m_2 + \cdots + m_k$. Then it is easily shown that

$$\chi_T = \chi_{T_1} + \chi_{T_2} + \cdots + \chi_{T_k}.$$

We have the following facts as consequences of this observation, and Proposition 3.3.3.

Proposition 3.3.10. Suppose $\chi : G \to F$ is a character of a group G.

(i) χ is completely reducible if and only if χ is a sum of isotypic characters.

(ii) χ is isotypic if and only if $\chi = m \cdot \psi$ for some integer $m > 0$ and some irreducible character ψ of G over F.

It is clear that we obtain partial information about a representation T from its character χ_T. We have noted that χ_T is irreducible (respectively, isotypic, or completely reducible), if T is irreducible (respectively, isotypic, or completely reducible). We have also noted that this information about the character of a representation is not generally sufficient to prove that the representation is isotypic, or completely reducible. We are interested in classes of representations T of groups G with the property that a representation is uniquely determined by its character, so that the representation T is completely reducible, isotypic, or irreducible if and only if its character is. With this in mind, we consider representations and characters of finite groups.

We first note that the problem of deciding whether a representation of a finite group G over a field F is completely reducible is trivial, for a large class of fields F.

Proposition 3.3.11. (Maschke's Theorem). Let $T : G \to GL(M, F)$ be a linear representation of a finite group G of size n, over a field F whose characteristic is either 0 or is positive and does not divide n; then T is completely reducible.

Proposition 3.3.11 is proved in Curtis and Reiner [31], or, assuming $F = \mathbb{C}$, in Serre [112] (see Theorem 1). The proof is constructive: it includes a (polynomial time) algorithm which takes as input a basis for a G-subspace N_1 of M, and produces a basis for a G-subspace N_2 of M, such that $M = N_1 \oplus N_2$. It is clear that Maschke's Theorem implies the analogous result for matrix representations; that is, that if G and F are above, then every matrix representation of G over F is completely reducible.

Applying Proposition 3.3.11 and Proposition 3.2.3, we obtain the following corollary.

Proposition 3.3.12. If G is a finite group with n elements, F is a field whose characteristic does not divide n, and T is a linear (or matrix) representation of G over F, then the associative algebras $\text{env}(T)$ and $\text{com}(T)$ are both semi-simple over F.

Note that Example 3.1.13 includes a matrix representation of the cyclic group of order n over $\mathbb{F}_p$, for p a prime divisor of n, which is not completely reducible (and whose enveloping algebra is not semi-simple) — demonstrating that the condition that the characteristic of F *not* divide $n = |G|$ is necessary for Maschke's Theorem.

We next show that there are only finitely many inequivalent irreducible representations of G over a field F whose characteristic does not divide $|G|$, and derive a (well known) upper bound for the number of inequivalent irreducible representations. Let $G = \{g_1, g_2, \ldots, g_n\}$ and recall the *regular matrix representation* $T_G : G \to GL(n, F)$ introduced in Example 3.1.3. We obtain a corresponding *regular linear representation* $\tilde{T}_G : G \to GL(M_G, F)$, for M_G an F-vector space of dimension n over F, with basis $\{ m_1, m_2, \ldots, m_n \}$ over F, by defining $\tilde{T}_G(g_i) \in GL(M_G, F)$ such that

$$\tilde{T}_G(g_i)(m_j) = (m_k) \qquad \text{for } k \text{ such that } g_i \cdot g_j = g_k \text{ in } G,$$

for $1 \le i, j \le n$. Then the matrix representation $T_G : G \to GL(n, F)$ corresponds to the linear representation $\tilde{T}_G$ and the basis $\{ m_1, m_2, \ldots, m_n \}$ for M_G over F. It is clear that the maps T_G and $\tilde{T}_G$ are both injective (group) homomorphisms, so the finite groups G, $T_G(G)$, and $\tilde{T}_G(G)$ are isomorphic. By Proposition 3.3.12, the enveloping algebras $\text{env}(T_G)$ and $\text{env}(\tilde{T}_G)$ are semi-simple. It is easily checked that these algebras are isomorphic to each other, and to the *group algebra* FG of Example 2.1.6. These observations are of use in proving the following fact.

Proposition 3.3.13. Let G and F be as above, and let T be an irreducible linear representation of G over F; then T is equivalent to an irreducible component of the regular linear representation $\tilde{T}_G$ of G over F.

Proposition 3.3.13 is a consequence of a more general result, stated (and proved) as Theorem 25.10 in [31].

Now since FG is semi-simple, the number of nonisomorphic left ideals of $FG \cong \text{env}(\tilde{T}_G)$, and hence the number of inequivalent irreducible components of $\tilde{T}_G$, is also the number of simple components of the semi-simple algebra FG. Thus we obtain the following corollary.

Proposition 3.3.14. Let G and F be as above; then the number of inequivalent irreducible linear (or matrix) representations of G over F is at most the number of simple components of the semi-simple algebra FG.

Since FG has dimension n over F, we can conclude immediately that there are at most n inequivalent irreducible representations. We obtain a tight bound for the number of inequivalent irreducible representations, for all groups over a general class of fields, by examining the structure of the algebra FG.

We noted in Section 2.4 that every simple component of a semi-simple algebra A (in particular, of FG) has an identity element which is a central idempotent in A. These central idempotents are clearly linearly independent in A over F (note that they annihilate each other). Hence, the number of simple components of A is at most the dimension over F of the centre of A.

Now we consider the centre of FG. Suppose $c_1g_1 + c_2g_2 + \cdots + c_ng_n \in \text{centre}(FG)$, for $c_1, c_2, \ldots, c_n \in F$. Then clearly

$$(c_1g_1 + c_2g_2 + \cdots + c_ng_n)g = g(c_1g_1 + c_2g_2 + \cdots + c_ng_n)$$

and

$$c_1(g^{-1}g_1g) + c_2(g^{-1}g_2g) + \cdots + c_n(g^{-1}g_ng) = c_1g_1 + c_2g_2 + \cdots + c_ng_n$$

for all $g \in G$. Since $g_1, g_2, \ldots, g_n$ are linearly independent in FG, it follows that $c_i = c_j$ if g_i and g_j are conjugates in G. Conversely, if $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_k$ are the conjugacy classes of G, then

$$\overline{\mathcal{C}_i} = \sum_{g \in \mathcal{C}_i} g \in \text{centre}(FG)$$

for $1 \leq i \leq k$. Clearly, $\overline{\mathcal{C}_1}, \overline{\mathcal{C}_2}, \ldots, \overline{\mathcal{C}_k}$ are linearly independent and span centre(FG) over F. We have proved the following.

Proposition 3.3.15. Let F and G be as above; then the number of simple components of FG over F is at most k, the number of conjugacy classes in G. Consequently, there are at most k inequivalent irreducible linear (or matrix) representations of G over F.

We saw in Section 2.4 that if A is semi-simple over F, with l simple components, then $\text{centre}(A) \cong E_1 \oplus E_2 \oplus \cdots \oplus E_l$, for finite algebraic extension fields $E_1, E_2, \ldots, E_l$ over F, so that the dimension of centre(A) is the sum of the dimensions of the fields E_i over F, for $1 \leq i \leq l$. If F is algebraically closed, then $E_i = F$ for all i, and it follows that $k = l$.

Proposition 3.3.16. Let F and G be as above, and suppose F is algebraically closed; then G has k inequivalent irreducible matrix (or linear) representations over F, where k is the number of conjugacy classes in G.

Thus there are at most k inequivalent irreducible matrix representations, and at most k distinct characters for G over F, if G is a finite group with n elements and k conjugacy classes, and n does not divide the characteristic of F. Since each character of G is a class function, so that it is specified by its value at the k conjugacy classes of G, we can specify all these values using a table with $l \leq k$ rows and k columns, with rows indexed by the l distinct characters of G over F, columns indexed by the conjugacy classes, and with the $(i, j)^{\text{th}}$ entry of the table equal to the value of the character indexing row i at an element of the conjugacy class indexing column j. We call such a table of values a *character table* for G over F.

Example 3.3.17. A character table for the dihedral group D_3 over the field $\mathbb{C}$ is included in Example 3.3.6.

We consider the problem stated on the following page. It is clear that we can solve this problem in polynomial time if we can form, and then decompose, a regular matrix representation of G at this cost. Given the input for this problem, the formation of a regular matrix representation of G is straightforward. Thus it follows from the results of Rónyai [102], [103] that we can compute a character table of a finite group G over a finite field $\mathbb{F}_{p^l}$ from a multiplication table for G, using a probabilistic algorithm, in polynomial time. We can also compute a character table for G over $\mathbb{C}$ or $\mathbb{R}$, with entries in a number field which is an extension with small degree over $\mathbb{Q}$, in (Boolean) probabilistic polynomial time (*cf.* Corollaries 3.2.10 and 3.2.11, and the remarks following Corollary 3.2.11).

In Sections 3.3.2–3.3.4 we discuss *deterministic* polynomial time algorithms for the computation of character tables over $\mathbb{C}$.

Problem **Computation of a Character Table**

Input.
- Integer $n > 0$, which does not divide the characteristic of F.
- A multiplication table for a finite group $G = \{ g_1, g_2, \ldots, g_n \}$ with n elements.

Output.
- The number k of conjugacy classes in G.
- The number l of distinct irreducible characters of G over the ground field F.
- A character table (with l rows and k columns) for G over F.

3.3.2. Burnside's Algorithm for Character Tables

We now assume G is a finite group with n elements, and that F is an algebraically closed field whose characteristic does not divide n — so that Maschke's Theorem is applicable, and G has exactly k inequivalent irreducible representations (and characters) over F, for k the number of conjugacy classes of G. (We will soon relax the restriction that F be algebraically closed: see Proposition 3.3.23 and the remarks which follow it.) We make the further assumption that if the characteristic of F is a positive prime p, then $p > 2n$. There is a well known algorithm, due to Burnside, which can be used under these conditions to compute the character table of G over F efficiently. We present and analyse Burnside's algorithm in this section.

We begin with a number of (well known) *orthogonality relations* for the irreducible characters of G, which are required to show that Burnside's algorithm is correct. These are stated as Proposition 3.3.18, below.

We will use the following notation. Once again, we let k be the number of conjugacy classes of G. We denote these conjugacy classes by $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_k$. For convenience, we assume $\mathcal{C}_1 = \{ 1_G \}$ (for 1_G the identity element in G). For $1 \le i \le k$, we denote by i^* the integer between 1 and k such that

$$\mathcal{C}_{i^*} = \{ g^{-1} \ : \ g \in \mathcal{C}_i \}.$$

We denote by h_i the number of elements of the conjugacy class $\mathcal{C}_i$, for $1 \le i \le k$, and we denote by $\zeta^{(1)}, \zeta^{(2)}, \ldots, \zeta^{(k)}$ the distinct irreducible characters of G over F, in some (arbitrarily chosen) order. Finally, we denote by z_i the dimension of an irreducible matrix representation of G over F with character $\zeta^{(i)}$. Since $1_G \in \mathcal{C}_1$, it is clear that $z_i = \zeta_1^{(i)}$ if F has characteristic 0, and that $\zeta_1^{(i)} = (z_i \bmod p)$ if F has positive characteristic p.

Proposition 3.3.18. Suppose G is a finite field with n elements, and that F is an algebraically closed field whose characteristic does not divide n. Let k, h_i, i^*, $\zeta^{(i)}$, z_i, and $\zeta_j^{(i)}$ be as defined above. Then

(i) $\displaystyle\sum_{g\in G} \zeta^{(i)}(hg)\zeta^{(j)}(g^{-1}) = \frac{\zeta^{(i)}(h)\cdot n}{z_i}\cdot \delta_{i\,j};$

(ii) $\displaystyle\sum_{g\in G} \zeta^{(i)}(g)\zeta^{(j)}(g^{-1}) = n\cdot \delta_{i\,j};$

(iii) $\displaystyle\sum_{l=1}^{k} h_l\zeta_l^{(i)}\zeta_{l^*}^{(j)} = n\cdot \delta_{i\,j};$

(iv) $\displaystyle\sum_{l=1}^{k} \zeta_i^{(l)}\zeta_{j^*}^{(l)} = \frac{n}{h_i}\cdot \delta_{i\,j};$

for $h\in G$, $1\le i,\ j\le k$ (and for $\delta_{i\,j}$ the Kronecker delta).

These relations are standard. For a proof of their correctness, see (for example) Section 31 of Curtis and Reiner [31]. They can be used to generate the character tables of some (very) small groups. For example, they are used to compute the character tables of the groups S_3, A_4, and S_4 (the symmetric group on 3 letters, and the alternating and symmetric groups on 4 letters) in Section 32 of Curtis and Reiner. In general, however, we must use some additional facts in order to compute character tables.

Definition 3.3.19. For $1\le r,\ s,\ t\le k$, set $c_{r\,s\,t}$ to be the number of solutions $(x,\ y)$ for the equation $x\cdot y = z$ with $x\in\mathcal{C}_r$, $y\in\mathcal{C}_s$, for some fixed $z\in\mathcal{C}_t$. It is easily checked that $c_{r\,s\,t}$ is independent of the choice of z, and that

$$\overline{\mathcal{C}_r}\cdot\overline{\mathcal{C}_s} = \sum_{t=1}^{k} c_{r\,s\,t}\overline{\mathcal{C}_t} \qquad (3.3.1)$$

in FG (with $c_{r\,s\,t}$ viewed as an element of the prime field of F). We call the values $c_{r\,s\,t}$ (for $1\le r,s,t\le k$) a set of *structure constants* for the group G.

Let $Z^{(j)}: G\to GL(z_j,F)$ be an irreducible matrix representation over F with character $\zeta^{(j)}$, for $1\le j\le k$. We have noted that $\overline{\mathcal{C}_i}\in$ Centre(FG), for $1\le j\le k$; consequently, $Z^{(j)}(\overline{\mathcal{C}_i})$ commutes with $Z^{(j)}(g)$ for all $g\in G$, and hence $Z^{(j)}(\overline{\mathcal{C}_i})\in \mathrm{com}(Z^{(j)})$. Since $Z^{(j)}$ is irreducible, $\mathrm{com}(Z^{(j)})$ is a finite-dimensional division algebra over F, by Proposition 3.2.5(a). Since F is an algebraically closed field, it follows that $\mathrm{com}(Z^{(j)}) \cong F$, so

$$Z^{(j)}(\overline{\mathcal{C}_i}) = \omega_i^{(j)} I_{z_j} \qquad \text{for some } \omega_i^{(j)} \in F; \tag{3.3.2}$$

that is, $Z^{(j)}(\overline{\mathcal{C}_i})$ is a scalar multiple of the identity matrix in $GL(z_j, F)$. Since the representation $Z^{(j)}$ has character $\zeta^{(j)}$ and $\zeta^{(j)}(g) = \zeta_i^{(j)}$ for all $g \in \mathbb{C}_i$, and since $\overline{\mathcal{C}_i} = \sum_{g \in \mathcal{C}_i} g$ in FG, and (finally) since $|\mathcal{C}_i| = h_i$, it follows that

$$z_j \omega_i^{(j)} = \mathrm{Trace}(Z^{(j)}(\overline{\mathcal{C}_i})) = \zeta^{(j)}(\overline{\mathcal{C}_i}) = h_i \cdot \zeta_i^{(j)};$$

thus

$$\omega_i^{(j)} = \frac{h_i \cdot \zeta_i^{(j)}}{z_j}, \qquad \text{for } 1 \leq i,\, j \leq k. \tag{3.3.3}$$

Now we apply the map $Z^{(j)}$ to Equation 3.3.1:

$$Z^{(j)}(\overline{\mathcal{C}_r}) \cdot Z^{(j)}(\overline{\mathcal{C}_s}) = \sum_{t=1}^{k} c_{r\,s\,t} Z^{(j)}(\overline{\mathcal{C}_t}),$$

or, using Equation 3.3.2,

$$\omega_r^{(j)} I_{z_j} \cdot \omega_s^{(j)} I_{z_j} = \sum_{t=1}^{k} c_{r\,s\,t} \omega_t^{(j)} \cdot I_{z_j}.$$

Thus,

$$\omega_r^{(j)} \cdot \omega_s^{(j)} = \sum_{t=1}^{k} \omega_t^{(j)}, \qquad \text{for } 1 \leq r,\, s \leq k. \tag{3.3.4}$$

Now if we denote by V_s the matrix in $M_{k \times k}(F)$ whose $(r,\, s)^{\text{th}}$ entry is $c_{r\,s\,t}$ for $1 \leq r,\, t \leq k$, then the above equation implies that

$$\omega_s^{(j)} \cdot \begin{bmatrix} \omega_1^{(j)} \\ \omega_2^{(j)} \\ \vdots \\ \omega_k^{(j)} \end{bmatrix} = V_s \cdot \begin{bmatrix} \omega_1^{(j)} \\ \omega_2^{(j)} \\ \vdots \\ \omega_k^{(j)} \end{bmatrix}, \tag{3.3.5}$$

so that the values of the characters $\zeta^{(1)}, \zeta^{(2)}, \ldots, \zeta^{(k)}$ at any element of the conjugacy class $\mathcal{C}_s$ are eigenvalues of the matrix V_s, while the vectors

$$w_1 = \begin{bmatrix} \omega_1^{(1)} \\ \omega_2^{(1)} \\ \vdots \\ \omega_k^{(1)} \end{bmatrix}, \quad w_2 = \begin{bmatrix} \omega_1^{(2)} \\ \omega_2^{(2)} \\ \vdots \\ \omega_k^{(2)} \end{bmatrix}, \quad \cdots, \quad w_k = \begin{bmatrix} \omega_1^{(k)} \\ \omega_2^{(k)} \\ \vdots \\ \omega_k^{(k)} \end{bmatrix}$$

are common eigenvalues of the matrices $V_1, V_2, \ldots, V_k$.

Burnside's algorithm is based on the above facts. The structure constants $c_{r\,s\,t}$ (for $1 \le r,\ s,\ t \le k$) for G are easily computed from the multiplication table for G; they are used (as entries of the matrices $V_1, V_2, \ldots, V_k$, whose eigenspaces are computed) to generate the vectors $w_1, w_2, \ldots, w_k$. Before stating the algorithm in detail, we show that $w_1, w_2, \ldots, w_k$ can be uniquely determined as common eigenvectors of $V_1, V_2, \ldots, V_k$.

We have defined $h_1, h_2, \ldots, h_k$ to be the sizes of the classes $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_k$ of G. Suppose $x \in \mathcal{C}_i$, and consider the set

$$H_x = \{\, y \in G \ : \ x \cdot y = y \cdot x \,\}.$$

It is easily checked that H_x is a subgroup of G, and that for any elements α, β of G, $\alpha x \alpha^{-1} = \beta x \beta^{-1}$ if and only if $\alpha \in \beta H_x = \{\, \beta y \ : \ y \in H_x \,\}$. Now if $x, y \in \mathcal{C}_i$ then it is easily checked that $|H_x| = |H_y|$ (if $y = \alpha x \alpha^{-1}$, then the map $\phi : g \mapsto \alpha g \alpha^{-1}$ defines a bijection from H_x to H_y). It follows that, if we denote by c_i the number of elements in H_x for any $x \in \mathcal{C}_i$, then

$$n = |G| = |\mathcal{C}_i| \cdot |H_x| = h_i \cdot c_i,$$

and, in particular, that h_i is a divisor of n. Since the characteristic of F does not divide n, it does not divide h_i either. Jacobson [68] notes that this is also the case for the integers $z_1, z_2, \ldots, z_k$; that is, if F is algebraically closed and has characteristic $p > 0$ such that p does not divide n, then p does not divide any z_i (see Section 5.5, Theorem 5.10, of Jacobson [68]). These facts are used to prove the following lemma.

Lemma 3.3.20. Let G, n, F, and the values $c_{r\,s\,t}$, V_s, and w_s be as defined above, for $1 \le r,\ s,\ t \le k$. For every pair of vectors w_i and w_j, with $1 \le i < j \le k$, there exists an integer l such that $1 \le l \le k$ and such that w_i and w_j are vectors in *distinct* eigenspaces of the matrix V_l.

Proof. We begin by showing that the vectors $w_1, w_2, \ldots, w_k$ are linearly independent over F. Let $W \in M_{k \times k}(F)$ be the matrix with i^{th} column w_i, for $1 \le i \le k$. Then

$$W = \begin{bmatrix} \omega_1^{(1)} & \omega_2^{(1)} & \cdots & \omega_k^{(1)} \\ \omega_1^{(2)} & \omega_2^{(2)} & \cdots & \omega_k^{(2)} \\ \vdots & \vdots & \ddots & \vdots \\ \omega_1^{(k)} & \omega_2^{(k)} & \cdots & \omega_k^{(k)} \end{bmatrix},$$

and it is clear that the vectors $w_1, w_2, \ldots, w_k$ are linearly independent over F if and only if the matrix W is nonsingular. Now, since

$$\omega_i^{(j)} = \frac{h_i \cdot \zeta_i^{(j)}}{z_j} \qquad \text{for } 1 \leq i, j \leq k,$$

it is clear that

$$W = \begin{bmatrix} z_1 & & & 0 \\ & z_2 & & \\ & & \ddots & \\ 0 & & & z_k \end{bmatrix}^{-1} \cdot \begin{bmatrix} \zeta_1^{(1)} & \zeta_2^{(1)} & \cdots & \zeta_k^{(1)} \\ \zeta_1^{(2)} & \zeta_2^{(2)} & \cdots & \zeta_k^{(2)} \\ \vdots & \vdots & \ddots & \vdots \\ \zeta_1^{(k)} & \zeta_2^{(k)} & \cdots & \zeta_k^{(k)} \end{bmatrix} \cdot \begin{bmatrix} h_1 & & & 0 \\ & h_2 & & \\ & & \ddots & \\ 0 & & & h_k \end{bmatrix}.$$

Since all of the integers $z_1, z_2, \ldots, z_k$ and $h_1, h_2, \ldots, h_k$ are positive, and none of them divide the characteristic of F, it is clear that the first and third of the matrices on the right side of this equation are nonsingular. It is a consequence of the orthogonality relations (stated in Proposition 3.3.18) that the rows of the second matrix are linearly independent, so that this matrix is nonsingular, as well. Thus W is nonsingular, and $w_1, w_2, \ldots, w_k$ are linearly independent over F.

Now suppose the lemma is false — so there exist vectors w_i and w_j, for i and j such that $1 \leq i < j \leq k$, such that w_i and w_j are in the same eigenspace of the matrix V_s, for all s between 1 and k. It follows, by Equation 3.3.5, that $\omega_s^{(i)} = \omega_s^{(j)}$ for all s, since these are the eigenvalues for the eigenvectors w_i and w_j for V_s. Thus the i^{th} and j^{th} columns of the matrix W are the same, contradicting the fact that W is nonsingular, as is required to prove the lemma. ∎

It follows that we can compute the vectors $w_1, w_2, \ldots, w_k$ to within constant factors, by computing bases of eigenspaces of the matrices $V_1, V_2, \ldots, V_k$, then using a divide and conquer approach, computing intersections of these subspaces (discarding empty subspaces, and subspaces for which proper subspaces have been generated); we will obtain k one-dimensional subspaces of $M_{k\times 1}(F)$. Choosing nonzero entries of these subspaces, we obtain vectors

$$\hat{w}_1 = \alpha_1 w_1, \quad \hat{w}_2 = \alpha_2 w_2, \quad \ldots, \quad \hat{w}_k = \alpha_k w_k,$$

for nonzero values $\alpha_1, \alpha_2, \ldots, \alpha_k$ in F. Since the ordering in which the characters $\zeta^{(1)}, \zeta^{(2)}, \ldots, \zeta^{(k)}$ are listed is arbitrary, we can choose the ordering at this point (by ordering the vectors $\hat{w}_1, \hat{w}_2, \ldots, \hat{w}_k$).

Now it remains only to eliminate the constants α_i, and compute the vectors $z_1, z_2, \ldots, z_k$, in order to obtain the values

$$\zeta_i^{(j)} = \frac{z_j \cdot \omega_i^{(j)}}{h_i} \qquad \text{for } 1 \leq i, j \leq k.$$

We eliminate $\alpha_1, \alpha_2, \ldots, \alpha_k$ by noting that, since $\mathcal{C}_1 = \{1\}$,

$$\zeta_1^{(i)} = \zeta^{(i)}(1) = \text{Trace}(Z^{(i)}(1)) = z_i.$$

Thus

$$z_i = \zeta_1^{(i)} = \frac{z_i \cdot \omega_1^{(i)}}{h_1},$$

and, since $z_i > 0$, and $h_1 = 1$, it follows that $\omega_i^{(i)} = 1$ for $1 \leq i \leq k$; α_i is the first entry of the vector

$$\hat{w}_i = \alpha_i \cdot \begin{bmatrix} \omega_1^{(i)} \\ \omega_2^{(i)} \\ \vdots \\ \omega_k^{(i)} \end{bmatrix} = \alpha_i \cdot \begin{bmatrix} 1 \\ \omega_2^{(i)} \\ \vdots \\ \omega_k^{(i)} \end{bmatrix}.$$

Now it remains only to compute $z_1, z_2, \ldots, z_k$ in order to recover the value $\zeta_i^{(j)}$, for $1 \leq i, j \leq k$. At this point we use the third of the orthogonality relations stated in Proposition 3.3.18. Using the values h_i, i^*, and $\omega_j^{(i)}$, which we have obtained, we compute the values S_i, for $1 \leq i \leq k$, given below.

$$\begin{aligned} S_i &= \sum_{l=1}^{k} \frac{1}{h_l} \cdot \omega_l^{(i)} \cdot \omega_{l^*}^{(i)} \\ &= \sum_{l=1}^{k} \left(\frac{h_l \left(\frac{z_i \cdot \omega_l^{(i)}}{h_l} \right) \cdot \left(\frac{z_i \cdot \omega_{l^*}^{(i)}}{h_{l^*}} \right)}{z_i^2} \right) && \text{since } h_l = h_{l^*} \\ &= \frac{1}{z_i^2} \sum_{l=1}^{k} \left(h_l \zeta_l^{(i)} \zeta_{l^*}^{(i)} \right) \\ &= \frac{n}{z_i^2}, && \text{by Proposition 3.3.18(iii).} \end{aligned}$$

Now $(n/S_i) = z_i^2 \in F$. Since z_i is the dimension of an irreducible matrix representation of G, which is a component of the regular matrix representation of G over F, $0 < z_i < n$. We now use our assumption that $p > 2n$ (if $p = \text{char}(F) > 0$) to conclude that the integer z_i can be determined from the value $(n/S_i) \in F$.

We state the resulting algorithm on the following page.

Algorithm **Character Table I (Burnside)**

Input.
- Integer $n > 0$.
- A multiplication table for a finite group $G = \{ g_1, g_2, \ldots, g_n \}$ with n elements.

Output.
- The number k of conjugacy classes of G.
- The number l of distinct irreducible characters of G, over the ground field F, an algebraically closed field of characteristic 0 or of characteristic p, for $p > 2n$ such that $\gcd(p, n) = 1$.
- A character table (with l rows and k columns) for G over F.

(1) Use the multiplication table for G to compute the number k of conjugacy classes, and the sizes $h_1, h_2, \ldots, h_k$ of these classes, $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_k$ (listed in some order), and to compute the number $c_{r\,s\,t}$ of solutions (x, y) of the equation $x \cdot y = z$ for fixed $z \in \mathcal{C}_t$, such that $x \in \mathcal{C}_r$ and $y \in \mathcal{C}_s$, for $1 \le r, s, t \le k$. Set $l = k$.

For $1 \le i \le k$, compute the integer i^* such that $g^{-1} \in \mathcal{C}_i$ if $g \in \mathcal{C}_i$, for $g \in G$.

(2) For $1 \le s \le k$, let $V_s \in M_{k\times k}(F)$ such that $(V_s)_{r\,t} = c_{r\,s\,t}$ for $1 \le r, t \le k$.

Compute the eigenvalues, and bases for the eigenspaces, of the matrices $V_1, V_2, \ldots, V_k$. Compute bases for intersections of these eigenspaces (using a divide and conquer approach) to obtain vectors

$$w_1 = \begin{bmatrix} \omega_1^{(1)} \\ \omega_2^{(1)} \\ \vdots \\ \omega_k^{(1)} \end{bmatrix}, \quad w_2 = \begin{bmatrix} \omega_1^{(2)} \\ \omega_2^{(2)} \\ \vdots \\ \omega_k^{(2)} \end{bmatrix}, \quad \ldots, \quad w_k = \begin{bmatrix} \omega_1^{(k)} \\ \omega_2^{(k)} \\ \vdots \\ \omega_k^{(k)} \end{bmatrix}$$

in $M_{k\times 1}(F)$, such that these are common eigenvectors of $V_1, V_2, \ldots, V_k$, span $M_{k\times 1}(F)$ over F, and such that $\omega_1^{(i)} = 1$ for $1 \le i \le k$.

(3) For $1 \le i \le k$, compute $S_i = \sum_{l=1}^{k} \frac{1}{h_l} \omega_l^{(i)} \omega_{l^*}^{(i)}$. Set z_i to be the unique integer such that $z_i > 0$ (if F has characteristic 0) or $0 < z_i < (p/2)$ (if F has positive characteristic p) and such that $S_i \cdot z_i^2 = n$ (in F).

(4) For $1 \le r, s \le k$, set the $(r, s)^{\text{th}}$ entry of the character table to be

$$\zeta_s^{(r)} = \frac{z_r \cdot \omega_s^{(r)}}{h_s} \in F.$$

Return the desired values.

In order to prove that we can implement this algorithm efficiently using Boolean circuits, we note that all values computed lie in a single small algebraic extension of the *prime field* of F (that is, of $\mathbb{Q}$ if F has characteristic 0, or of $\mathbb{F}_p$, if F has characteristic p, for $p > 0$). We define the *exponent* of G, $\exp(G)$, to be the lowest common multiple of the orders of the elements of G. In general, $\exp(G)$ divides n. If G is cyclic then $\exp(G) = n$; however, $\exp(G)$ can be much smaller than n.

Example 3.3.21. The additive group $\mathbb{F}_2^k = \mathbb{F}_2 \oplus \mathbb{F}_2 \oplus \cdots \oplus \mathbb{F}_k$ (with k summands) has order $n = 2^k$ and exponent 2, for all $k > 0$.

Definition 3.3.22. Let G be a finite group. A field F is a *splitting field* for G if, for every irreducible matrix representation $\hat{T}$ for G over F and every field extension E of F, the representation $\hat{T}_E$ is irreducible over E.

Proposition 3.3.23. Let G be a finite group with order n and exponent m.

(i) If F is a field whose characteristic is zero or is positive and does not divide n, and if F contains an m^{th} primitive root of unity, then F is a splitting field for G.

(ii) If F is as in (i) and E is an algebraic closure of F then a character table for G over F is also a character table for G over E. Furthermore, the entries of a character table for G over F all lie in $F \cap K[\omega]$, for K the prime field of F and for ω an m^{th} primitive root of unity in E. If F has characteristic zero, so that $K = \mathbb{Q}$, then the entries of a character table for G over F are all algebraic integers.

Part of Proposition 3.3.23 follows from the fact that $g^m = 1$ for all $g \in G$. Thus, if T is a matrix representation for G over F, then $(T(g))^m = 1$, so the characteristic values of the matrix $T(g)$ are all m^{th} roots of unity in E. Hence the entries of a character table for G over F are all members of the subfield $K[\omega]$ of E, as well as of F, and are algebraic integers if $K = \mathbb{Q}$.

It is more difficult to show that any matrix representation of G over E is equivalent to some matrix representation of G over F — and also to some matrix representation of G over $K[\omega]$. (Note that this is also implied by the above proposition.) This was conjectured by Maschke, and first proved by Brauer; for a proof of this result, and additional references, see Sections 41 and 70 of [31] (see in particular Theorems 41.1 and 70.23).

We can now relax our earlier restriction that F be algebraically closed: It is sufficient to assume instead that F is a splitting field for G, which contains an m^{th} primitive root of unity. We will continue to assume also that the characteristic of F is either zero or greater than $2n$.

Now the structure constants $c_{r\,s\,t}$ are integers between 0 and n, for $1 \leq r,\ s,\ t \leq k$; they each have a binary representation of length $O(\log n)$. It is easily checked that the intermediate values generated by Burnside's algorithm each have Boolean representations with length polynomial in n. Applying the results for factorisation and linear algebra summarised in Section 1, it is easily checked that Burnside's algorithm can be implemented to compute character tables of finite groups over $\mathbb{C}$ efficiently. Thus, we have proved the following result.

Theorem 3.3.24. Let G be a finite group of order n, and let F be a splitting field for G of characteristic 0. Then the character table of G over $\mathbb{C}$ can be computed from a multiplication table for G, using a family of Boolean circuits of size polynomial in n.

Note that, if the splitting field F (of the above theorem) does not include an m^{th} primitive root of unity, then since m does not divide the characteristic of F, some algebraic extension of small degree over F includes such a root; we can compute a character table for G over F by working in a (small) extension, and then recovering (binary) representations of the entries of the table as elements of the ground field F.

Example 3.3.25. We will use Burnside's method to compute the character table of D_3 over $\mathbb{C}$. We use as input the following multiplication table for D_3.

g:	1	a	a^2	b	ab	a^2b
$1 \cdot g$:	1	a	a^2	b	ab	a^2b
$a \cdot g$:	a	a^2	1	ab	a^2b	b
$a^2 \cdot g$:	a^2	1	a	a^2b	b	ab
$b \cdot g$:	b	a^2b	ab	1	a^2	a
$ab \cdot g$:	ab	b	a^2b	a	1	a^2
$a^2b \cdot g$:	a^2b	ab	b	a^2	a	1

Checking values $h^{-1}gh$ for $g,\ h \in G$, we group the elements of G into three conjugacy classes $\mathcal{C}_1$, $\mathcal{C}_2$, and $\mathcal{C}_3$, with orders h_1, h_2, and h_3 respectively, and compute integers 1^*, 2^*, and 3^*:

$$\begin{array}{lll}
\mathcal{C}_1 = \{\,1\,\}, & \mathcal{C}_2 = \{\,a,\ a^2\,\}, & \mathcal{C}_3 = \{\,b,\ ab,\ ab^2\,\}; \\
h_1 = 1, & h_2 = 2, & h_3 = 3; \\
1^* = 1, & 2^* = 2, & 3^* = 3.
\end{array}$$

Elements of the classes $\mathcal{C}_1$, $\mathcal{C}_2$, and $\mathcal{C}_3$ have orders 1, 3, and 2 respectively; hence the exponent of G is $m = \operatorname{lcm}(1,\ 3,\ 2) = 6$. We will express the entries of the character table as elements of the algebraic extension $\mathbb{Q}[\omega]$ of $\mathbb{Q}$, for ω a 6^{th} primitive root of unity. As indicated in Section 1, we identify ω (and the field $\mathbb{Q}[\omega]$) by stating the minimal polynomial of ω over $\mathbb{Q}$, as well as an isolating region for ω in $\mathbb{C}$. The minimal polynomial for an m^{th} primitive root of unity over $\mathbb{Q}$ is the m^{th} *cyclotomic polynomial,* Ψ_m; the coefficients of this polynomial are easily computed using the formula

$$\Psi_m = \prod_{d \text{ divides } m} (x^d - 1)^{\mu(m/d)},$$

where $\mu(n)$ is the *Möbius function,* given for positive integers n by the rules $\mu(1) = 1$, $\mu(p_1 p_2 \cdots p_r) = (-1)^r$ if p_1, p_2, $\ldots$, p_r are distinct primes, and $\mu(n) = 0$ if $n > 0$ is not squarefree. For this example, ω has minimal polynomial

$$\Psi_6 = (x^6 - 1)(x^3 - 1)^{-1}(x^2 - 1)^{-1}(x - 1) = x^2 - x + 1.$$

In general, we can choose as ω any root of Ψ_m — including the root

$$e^{\frac{2\pi\sqrt{-1}}{m}} = \cos\left(\frac{2\pi}{m}\right) + \sqrt{-1} \cdot \sin\left(\frac{2\pi}{m}\right).$$

We can compute an isolating region for this root using standard (Taylor series) approximations for the trigonometric functions, sine and cosine. For our example, we can use

$$\omega = \cos\left(\frac{\pi}{3}\right) + \sqrt{-1} \cdot \sin\left(\frac{\pi}{3}\right) = \frac{1}{2} + \sqrt{-1} \cdot \frac{\sqrt{3}}{2};$$

we can use the isolating region given by the inequalities

$$0.4 < \mathrm{Re}(\omega) < 0.6, \qquad \text{and} \qquad 0.8 < \mathrm{Im}(\omega) < 0.9.$$

Each entry $\zeta_j^{(i)}$ of the character table will be specified by values $\zeta_{j,0}^{(i)}$, $\zeta_{j,1}^{(i)} \in \mathbb{Q}$ such that

$$\zeta_j^{(i)} = \zeta_{j\,0}^{(i)} + \zeta_{j\,1}^{(i)} \cdot \omega.$$

Now since G has 3 conjugacy classes, the character table to be computed will have 3 rows and 3 columns. By inspection of the multiplication table (and evaluation of products $x \cdot y$ for $x \in \mathcal{C}_r$ and $y \in \mathcal{C}_s$, for $1 \leq r,\ s \leq 3$), we obtain the following structure constants for the group G (and for the conjugacy classes $\mathcal{C}_1$, $\mathcal{C}_2$, and $\mathcal{C}_3$ indicated above).

r:	1	2	3	r:	1	2	3	r:	1	2	3
$c_{r\,1\,1}$:	1	0	0	$c_{r\,2\,1}$:	0	2	0	$c_{r\,3\,1}$:	0	0	3
$c_{r\,1\,2}$:	0	1	0	$c_{r\,2\,2}$:	1	1	0	$c_{r\,3\,2}$:	0	0	3
$c_{r\,1\,3}$:	0	0	1	$c_{r\,2\,3}$:	0	0	2	$c_{r\,3\,3}$:	1	2	0

Thus we obtain vectors w_1, w_2, $w_3 \in M_{3\times 1}(\mathbb{Q}[\omega]) \subseteq M_{3\times 1}(\mathbb{C})$ as common eigenvectors (with first entry 1) of the matrices

$$V_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \qquad V_2 = \begin{bmatrix} 0 & 1 & 0 \\ 2 & 1 & 0 \\ 0 & 0 & 2 \end{bmatrix}, \qquad V_3 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 2 \\ 3 & 3 & 0 \end{bmatrix}.$$

Computing and factoring the characteristic polynomials (over $\mathbb{Q}[\omega]$), we find that the characteristic polynomials of V_1, V_2, and V_3 are, respectively,

$$\begin{aligned} \phi_1 &= t^3 - 3t^2 + 3t - 1 = (t-1)^3, \\ \phi_2 &= t^3 - 3t^2 + 4 = (t-2)^2(t+1), \text{ and} \\ \phi_3 &= t^3 - 6t^2 - 3t = t(t+3)(t-3). \end{aligned}$$

Since ϕ_3 is squarefree, we can use V_3 alone to compute the vectors w_1, w_2, and w_3, as eigenvalues for the characteristic values 0, 3, and -3 respectively.

Solving systems of linear equations over $\mathbb{Q}[\omega]$, we see that, for $w \in M_{3\times 1}(\mathbb{Q}[\omega])$,

$$V_3 \cdot w = 0 \text{ if and only if } w = \alpha \cdot \begin{bmatrix} 1 \\ -1 \\ 0 \end{bmatrix} \text{ for some } \alpha \in \mathbb{Q}[\omega];$$

$$V_3 \cdot w = 3w \text{ if and only if } w = \alpha \cdot \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} \text{ for some } \alpha \in \mathbb{Q}[\omega]; \text{ and}$$

$$V_3 \cdot w = -3w \text{ if and only if } w = \alpha \cdot \begin{bmatrix} 1 \\ 2 \\ -3 \end{bmatrix} \text{ for some } \alpha \in \mathbb{Q}[\omega].$$

Thus (since $w_1^{(i)} = 1$ for all i), we have

$$w_1 = \begin{bmatrix} 1 \\ -1 \\ 0 \end{bmatrix}, \qquad w_2 = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}, \qquad w_3 = \begin{bmatrix} 1 \\ 2 \\ -3 \end{bmatrix}, \qquad \text{and}$$

$$\begin{array}{lll} \omega_1^{(1)} = 1 & \omega_1^{(2)} = 1 & \omega_1^{(3)} = 1 \\ \omega_2^{(1)} = -1 & \omega_2^{(2)} = 2 & \omega_2^{(3)} = 2 \\ \omega_3^{(1)} = 0 & \omega_3^{(2)} = 3 & \omega_3^{(3)} = -3. \end{array}$$

We next compute the values S_1, S_2, and S_3:

$$\begin{aligned}
S_1 &= \frac{1}{h_1}\omega_1^{(1)}\omega_{1^*}^{(1)} + \frac{1}{h_2}\omega_2^{(1)}\omega_{2^*}^{(1)} + \frac{1}{h_3}\omega_3^{(1)}\omega_{3^*}^{(1)} \\
&= \frac{1}{1}\cdot 1\cdot 1 + \frac{1}{2}\cdot(-1)\cdot(-1) + \frac{1}{3}\cdot 0\cdot 0 = \frac{3}{2}; \\
S_2 &= \frac{1}{h_1}\omega_1^{(2)}\omega_{1^*}^{(2)} + \frac{1}{h_2}\omega_2^{(2)}\omega_{2^*}^{(2)} + \frac{1}{h_3}\omega_3^{(2)}\omega_{3^*}^{(2)} \\
&= \frac{1}{1}\cdot 1\cdot 1 + \frac{1}{2}\cdot 2\cdot 2 + \frac{1}{3}\cdot 3\cdot 3 = 6; \\
S_3 &= \frac{1}{h_1}\omega_1^{(3)}\omega_{1^*}^{(3)} + \frac{1}{h_2}\omega_2^{(3)}\omega_{2^*}^{(3)} + \frac{1}{h_3}\omega_3^{(3)}\omega_{3^*}^{(3)} \\
&= \frac{1}{1}\cdot 1\cdot 1 + \frac{1}{2}\cdot 2\cdot 2 + \frac{1}{3}\cdot(-3)\cdot(-3) = 6.
\end{aligned}$$

Thus

$$\begin{aligned}
z_1^2 &= \frac{n}{S_1} = \frac{6}{(3/2)} = 4, \text{ and } z_1 = 2; \\
z_2^2 &= \frac{n}{S_2} = \frac{6}{6} = 1, \text{ and } z_2 = 1; \text{ and} \\
z_3^2 &= \frac{n}{S_3} = \frac{6}{6} = 1, \text{ and } z_3 = 1.
\end{aligned}$$

Finally, we compute and return the entries

$$\begin{array}{lll}
\zeta_1^{(1)} = \frac{z_1\omega_1^{(1)}}{h_1} = 2; & \zeta_2^{(1)} = \frac{z_1\omega_2^{(1)}}{h_2} = -1; & \zeta_3^{(1)} = \frac{z_1\omega_3^{(1)}}{h_3} = 0; \\
\zeta_1^{(2)} = \frac{z_2\omega_1^{(2)}}{h_1} = 1; & \zeta_2^{(2)} = \frac{z_2\omega_2^{(2)}}{h_2} = 1; & \zeta_3^{(2)} = \frac{z_2\omega_3^{(2)}}{h_3} = 1; \\
\zeta_1^{(3)} = \frac{z_3\omega_1^{(3)}}{h_1} = 1; & \zeta_2^{(3)} = \frac{z_3\omega_2^{(3)}}{h_2} = 1; & \zeta_3^{(3)} = \frac{z_3\omega_3^{(3)}}{h_3} = -1.
\end{array}$$

Note that this agrees with the set of values shown in Example 3.3.6, to within the ordering of the rows, as required.

Note that the entries of this character table are all integers — so the smallest field containing all entries of this table (namely, $\mathbb{Q}$) is a *proper* subfield of $\mathbb{Q}[\omega]$ in this case.

3.3.3. Dixon's Algorithm for Character Tables

It would appear that factorisation over $\mathbb{Q}$ or a number field is necessary, if we are to compute character tables over $\mathbb{C}$ — or, at least, that computations of character tables for finite groups with k conjugacy classes would not be less expensive than factorisation of polynomials of degree k over $\mathbb{Q}$. Surprisingly, this is not the case: In this section we present and analyse a modification of Burnside's algorithm suggested by Dixon [34], which allows us to replace factorisation of polynomials over $\mathbb{Q}$ in the algorithm by factorisation of polynomials over small finite fields.

Once again, suppose the finite group G has n elements, k conjugacy classes, and exponent m. Let p be a prime greater than $2n$, and let $F = \mathbb{F}_p[\hat{\omega}]$, for $\hat{\omega}$ an m^{th} primitive root of unity in an algebraic closure of $\mathbb{F}_p$. Suppose ω is an m^{th} primitive root of unity in $\mathbb{C}$; then there is a ring homomorphism

$$\phi : \mathbb{Z}[\omega] \to \mathbb{F}_p[\hat{\omega}]$$

such that $\phi(\omega) = \hat{\omega}$. We have seen that every entry $\zeta_j^{(i)}$ of a character table for G over $\mathbb{C}$ has the form

$$\zeta_j^{(i)} = \zeta_{j,0}^{(i)} + \zeta_{j,1}^{(i)}\omega + \cdots + \zeta_{j,m-1}^{(i)}\omega^{m-1},$$

for *integers* $\zeta_{j,0}^{(i)}, \zeta_{j,1}^{(i)}, \ldots, \zeta_{j,m-1}^{(i)}$. Clearly, then,

$$\begin{aligned}\phi(\zeta_j^{(i)}) &= \phi(\zeta_{j,0}^{(i)} + \zeta_{j,1}^{(i)}\omega + \cdots + \zeta_{j,m-1}^{(i)}\omega^{m-1})\\ &= \phi(\zeta_{j,0}^{(i)}) + \phi(\zeta_{j,1}^{(i)})\hat{\omega} + \cdots + \phi(\zeta_{j,m-1}^{(i)})\hat{\omega}^{m-1}.\end{aligned}$$

Dixon argues, for the special case $\hat{\omega} \in \mathbb{F}_p = F$, that the values $\phi(\zeta_j^{(i)})$ are entries of a character table for G over an algebraic closure of $\mathbb{F}_p$, so that these are the values computed by using Burnside's algorithm over such a field. We generalise his argument by removing the assumption that $\hat{\omega}$ belongs to the prime field $\mathbb{F}_p$. The details are given below.

Let $c_{r\,s\,t} \in \mathbb{Z}$ and let $V_1, V_2, \ldots, V_k \in M_{k\times k}(\mathbb{Z})$ be as defined in Section 3.3.2, and consider the vectors

$$w_1 = \begin{bmatrix}\omega_1^{(1)}\\ \omega_2^{(1)}\\ \vdots\\ \omega_k^{(1)}\end{bmatrix}, \qquad w_2 = \begin{bmatrix}\omega_1^{(2)}\\ \omega_2^{(2)}\\ \vdots\\ \omega_k^{(2)}\end{bmatrix}, \qquad \cdots, \qquad w_k = \begin{bmatrix}\omega_1^{(k)}\\ \omega_2^{(k)}\\ \vdots\\ \omega_k^{(k)}\end{bmatrix}.$$

These are eigenvectors of $V_1, V_2, \ldots, V_k$; hence, so are the vectors

$$z_1 w_1 = \begin{bmatrix} h_1\zeta_1^{(1)} \\ h_2\zeta_2^{(1)} \\ \vdots \\ h_k\zeta_k^{(1)} \end{bmatrix}, \qquad z_2 w_2 = \begin{bmatrix} h_1\zeta_1^{(2)} \\ h_2\zeta_2^{(2)} \\ \vdots \\ h_k\zeta_k^{(2)} \end{bmatrix}, \qquad \cdots, \qquad z_k w_k = \begin{bmatrix} h_1\zeta_1^{(k)} \\ h_2\zeta_2^{(k)} \\ \vdots \\ h_k\zeta_k^{(k)} \end{bmatrix}.$$

The entries of these vectors are in $\mathbb{Z}[\omega]$; they are determined, to within order of appearance and constant factors (in $\mathbb{C}$), as eigenvectors of the matrices $V_1, V_2, \ldots, V_k$ — and also as eigenvectors of $zV_1, zV_2, \ldots, zV_k$, for $z = \prod_{i=1}^k z_i$. The eigenvalues of the latter set of matrices are elements of $\mathbb{Z}[\omega]$; specifically, zV_s has eigenvalues

$$z\omega_s^{(j)} = h_s\zeta_s^{(j)} \prod_{\substack{l=1 \\ l\neq j}}^{k} z_l, \qquad \text{for } 1 \leq j \leq k.$$

Now consider the application of Burnside's algorithm, for the group G, over an algebraic closure of $\mathbb{F}_p$ containing $\hat{\omega}$. If we denote by $\hat{\zeta}_j^{(i)}$ the entries of a character table for G over this field (to distinguish them from the entries $\zeta_j^{(i)}$ of a character table for G over $\mathbb{C}$), then we can argue as above that these are determined, to within constant factors and order of appearance, by the fact that the vectors

$$\tilde{z}_1 \hat{w}_1 = \begin{bmatrix} h_1\hat{\zeta}_1^{(1)} \\ h_2\hat{\zeta}_2^{(1)} \\ \vdots \\ h_k\hat{\zeta}_k^{(1)} \end{bmatrix}, \qquad \tilde{z}_2 \hat{w}_2 = \begin{bmatrix} h_1\hat{\zeta}_1^{(2)} \\ h_2\hat{\zeta}_2^{(2)} \\ \vdots \\ h_k\hat{\zeta}_k^{(2)} \end{bmatrix}, \qquad \cdots, \qquad \tilde{z}_k \hat{w}_k = \begin{bmatrix} h_1\hat{\zeta}_1^{(k)} \\ h_2\hat{\zeta}_2^{(k)} \\ \vdots \\ h_k\hat{\zeta}_k^{(k)} \end{bmatrix}$$

are simultaneous eigenvectors of the matrices $\tilde{z}\hat{V}_1, \tilde{z}\hat{V}_2, \ldots, \tilde{z}\hat{V}_k$, and by the fact that $h_1\hat{\zeta}_1^{(i)} = \tilde{z}_i$, for $1 \leq i \leq k$, where $\tilde{z}_i$ is the dimension of the character $\hat{\zeta}^{(i)}$, and for $\tilde{z} = \prod_{i=1}^k \tilde{z}_i$ (extending ϕ to be a homomorphism from vectors and matrices with entries in $\mathbb{Z}[\omega]$ to vectors and matrices in $\mathbb{F}_p[\hat{\omega}]$, and setting $\hat{V}_s = \phi(V_s)$ for $1 \leq s \leq k$).

Now since $(zV_s)\cdot(z_i w_i) = (z\omega_s^{(i)})\cdot(z_i w_i)$ and the entries of these vectors and matrices are all in $\mathbb{Z}[\omega]$,

$$\phi(zV_s) \cdot \phi(z_i w_i) = \phi(z\omega_s^{(i)}) \cdot \phi(z_i w_i),$$

so that $\phi(z_i w_i)$ is an eigenvector of the matrix $\phi(zV_s)$, and of $\phi(V_s)$ (since $\phi(z) \neq 0$). We again use the fact that our ordering of the characters $\zeta^{(1)}, \zeta^{(2)}, \ldots, \zeta^{(k)}$ is completely arbitrary, as well as the fact that Burnside's algorithm is correct both over (algebraic closures of) $\mathbb{F}_p[\hat{\omega}]$ and $\mathbb{Q}[\omega]$, to conclude that

$$\phi(z_i w_i) = c_i \tilde{z}_i \hat{w}_i,$$

for some nonzero constants $c_1, c_2, \ldots, c_k \in \mathbb{F}_p[\hat{\omega}]$ and for $1 \leq i \leq k$. Thus, $\phi(h_j \zeta_j^{(i)}) = c_i h_j \hat{\zeta}_j^{(i)}$, and

$$\phi(\zeta_j^{(i)}) = c_i \hat{\zeta}_j^{(i)} \qquad \text{for } 1 \leq i,\, j \leq k.$$

Applying Proposition 3.3.18(ii), we see that

$$\begin{aligned} \phi(n) &= \phi\left(\sum_{g \in G} \zeta^{(i)}(g)\zeta^{(i)}(g^{-1})\right) \\ &= \sum_{g \in G} \phi(\zeta^{(i)}(g))\phi(\zeta^{(i)}(g^{-1})) \\ &= c_i^2 \sum_{g \in G} \hat{\zeta}^{(i)}(g)\hat{\zeta}^{(i)}(g^{-1}) \\ &= c_i^2 \phi(n), \end{aligned}$$

so that $c_i = \pm 1$ in $\mathbb{F}_p$ for all i. However,

$$\phi(z_i) = \phi(\zeta_1^{(i)}) = c_i \hat{\zeta}_1^{(i)} = c_i \tilde{z}_i;$$

since $\tilde{z}_i$ is the dimension of a component of the regular matrix representation of G, $\tilde{z}_i$ must be one of $1, 2, \ldots, n$. Now we use that assumption that $p > 2n$ to conclude that $\tilde{z}_i$ and $c_i \tilde{z}_i$ can both be in this range, for $c_i = \pm 1$, only if $c_i = 1$. We have proved the following lemma.

Lemma 3.3.26. Let G, n, k, and m be as above. Let p be a prime greater than $2n$, and let F be an algebraically closed field of characteristic p. Let ω be an m^{th} primitive root of unity in $\mathbb{C}$, and let $\hat{\omega}$ be an m^{th} primitive root of unity in F. Suppose values $\zeta_j^{(i)}$ (for $1 \leq i,\, j \leq k$) are the entries of a character table for G over $\mathbb{C}$; then $\zeta_j^{(i)} \in \mathbb{Z}[\omega]$, and the values $\phi(\zeta_j^{(i)})$ are entries of a character table for G over F, for $\phi : \mathbb{Z}[\omega] \to \mathbb{F}_p[\hat{\omega}]$ the homomorphism of rings taking ω to $\hat{\omega}$ (and taking each integer i to ($i \bmod p$)).

Since character tables are unique, up to the ordering of rows and columns, this lemma implies that *every* character table for G over F is the image under ϕ of a character table for G over $\mathbb{C}$. Dixon [34] includes a method for recovering a character table for $\mathbb{C}$ from its image under ϕ, for the case $\hat{\omega} \in \mathbb{F}_p$ (that is, for the case k divides $p - 1$, and the elements $\phi(\zeta_j^{(i)})$ all lie in the prime field of F). We present his method below, noting that it generalises directly to the more general case that the characteristic of F is any prime greater than $2n$.*

Once again, we recall that the entries of a character table for G over $\mathbb{C}$ all lie in $\mathbb{Z}[\omega]$, for ω an m^{th} primitive root of unity in $\mathbb{C}$:

$$\zeta_j^{(i)} = \zeta_{j,\,0}^{(i)} + \zeta_{j,\,1}^{(i)}\omega + \cdots + \zeta_{j,\,m-1}^{(i)}\omega^{m-1},$$

for integers $\zeta_{j,\,0}^{(i)}, \zeta_{j,\,1}^{(i)}, \ldots, \zeta_{j,\,m-1}^{(i)}$ between 0 and n — because each entry $\zeta_j^{(i)}$ is the trace of a matrix of order at most n, whose entries are all m^{th} roots of unity; $\zeta_{j,\,l}^{(i)}$ is the multiplicity of the root ω^l as a characteristic value of this matrix. Let s be an integer between 0 and $m-1$; since the characteristic values of the s^{th} power of a matrix are the s^{th} powers of the characteristic values of the original matrix,

$$\zeta_{\langle\, j,\, s\,\rangle}^{(i)} = \zeta_{j,\,0}^{(i)} + \zeta_{j,\,1}^{(i)}\omega^s + \zeta_{j,\,2}^{(i)}\omega^{2s} + \cdots + \zeta_{j,\,m-1}^{(i)}\omega^{(m-1)s},$$

for $\langle\, j,\, s\,\rangle$ the integer between 1 and k such that $g^s \in \mathcal{C}_{\langle\, j,\, s\,\rangle}$ if $g \in \mathcal{C}_j$ (note that $\langle\, j,\, s\,\rangle$ is independent of the choice of g in $\mathcal{C}_j$).

Now we use the fact that for any integer s such that $-m < s < m$ and $s \neq 0$, $1+\omega^s+\omega^{2s}+\cdots+\omega^{(m-1)s} = 0$ (since ω^s is a root of the polynomial $(t^m-1)/(t-1) = t^{m-1}+t^{m-2}+\cdots+t+1$), while $1+\omega^s+\omega^{2s}+\cdots+\omega^{(m-1)s} = m$ if m divides s. Thus, if s is between 0 and $m-1$, and $1 \le i,\, j \le k$,

$$\begin{aligned}
&\sum_{t=0}^{m-1} \zeta_{\langle\, j,\, t\,\rangle}^{(i)}\omega^{-st} \\
&\qquad= \sum_{t=0}^{m-1} \left(\zeta_{j,\,0}^{(i)} + \zeta_{j,\,1}^{(i)}\omega^t + \cdots + \zeta_{j,\,m-1}^{(i)}\omega^{t(m-1)}\right)\omega^{-st} \\
&\qquad= \sum_{t=0}^{m-1} \left(\zeta_{j,\,0}^{(i)}\omega^{-st} + \zeta_{j,\,1}^{(i)}\omega^{(1-s)t} + \cdots + \zeta_{j,\,m-1}^{(i)}\omega^{(m-1-s)t}\right) \\
&\qquad= \sum_{r=0}^{s-1} \zeta_{j,\,r}^{(i)}\left(\sum_{t=0}^{m-1}\omega^{(r-s)t}\right) + \zeta_{j,\,s}^{(i)}\left(\sum_{t=0}^{m-1}\omega^{0\cdot t}\right) + \sum_{r=s+1}^{m-1} \zeta_{j,\,r}^{(i)}\left(\sum_{t=0}^{m-1}\omega^{(r-s)t}\right) \\
&\qquad= \sum_{r=0}^{s-1} \zeta_{j,\,r}^{(i)}\cdot 0 + \zeta_{j,\,s}^{(i)}\cdot m + \sum_{r=s+1}^{m-1} \zeta_{j,\,r}^{(i)}\cdot 0 \\
&\qquad= m\cdot\zeta_{j,\,s}^{(i)}.
\end{aligned}$$

* In fact, Dixon includes smaller primes p; he assumes only that $p > 2\sqrt{n}$. We give a slightly weaker result, which is more easily proved.

Clearly, then,

$$\begin{aligned}\sum_{t=0}^{m-1} \phi(\zeta_{\langle j,t\rangle}^{(i)})\hat{\omega}^{-st} &= \sum_{t=0}^{m-1} \phi(\zeta_{\langle j,t\rangle}^{(i)})\hat{\omega}^{(m-s)t} \\ &= \phi\left(\sum_{t=0}^{m-1} \zeta_{\langle j,t\rangle}^{(i)}\omega^{(m-s)t}\right) \\ &= \phi(m \cdot \zeta_{j,s}^{(i)}) \\ &= \phi(m) \cdot \phi(\zeta_{j,s}^{(i)}), \qquad \text{as well.}\end{aligned}$$

Since $p > 2n$, and $\zeta_{j,s}^{(i)}$ is an integer between 0 and n, it is clear that $\zeta_{j,s}^{(i)}$ can be recovered from its image, $\phi(\zeta_{j,s}^{(i)})$; this image can be obtained using Burnside's algorithm over a field F of characteristic p, to compute the entries of the character table (mod p), then using sums of the above form to recover the images $\zeta_{j,s}^{(i)} \in \mathbb{F}_p$, for $1 \leq i, j \leq k$ and $0 \leq s < m$.

Dixon's original algorithm, and the new algorithm to be presented in Section 3.3.4, compute character tables for finite groups over $\mathbb{C}$ by choosing a suitable finite field, computing a character table over an algebraic closure of that finite field, and then recovering a character table over $\mathbb{C}$ as described above. The algorithms differ in the type of finite field which is used.

We first consider Dixon's algorithm. The finite field used in this version presented here is a prime field $\mathbb{F}_p$, with characteristic greater than $2n$, and which includes an m^{th} primitive root of unity. (We note again that Dixon uses the lower bound $2\sqrt{n}$, instead of $2n$, for the characteristic.) Since the multiplicative subgroup of $\mathbb{F}_p$ is cyclic, it is clear that $\mathbb{F}_p$ is a suitable finite field for any prime p such that

$$p > 2n \qquad \text{and} \qquad p \equiv 1 \pmod{m}.$$

It is sufficient, then, to choose any prime greater than $2n$ from the arithmetic progression

$$m+1,\ 2m+1,\ 3m+1,\ 4m+1,\ \ldots$$

Alternatively, we can choose any prime from the progression

$$M+1,\ 2M+1,\ 3M+1,\ 4M+1,\ \ldots$$

where $M = 2n$; then m divides M. It is clear that we must show that such progressions include small primes, if we are to prove that Dixon's algorithm, which we state on the following pages, is efficient.

Algorithm **Character Table II (Dixon)**

Input.
- Integer $n > 0$.
- A multiplication table for a finite group $G = \{ g_1, g_2, \ldots, g_n \}$ with n elements.

Output.
- The number k of conjugacy classes of G. (Note that k is also the number of distinct irreducible characters of G over $\mathbb{C}$.)
- An integer $m > 0$, such that the entries of a character table for G over $\mathbb{C}$ all lie in $\mathbb{Q}[\omega]$, for ω an m^{th} primitive root of unity.
- A character table for G over $\mathbb{C}$, with k rows and columns, and with each entry $\zeta_j^{(i)}$ given by integers $\zeta_{j,0}^{(i)}, \zeta_{j,1}^{(i)}, \ldots, \zeta_{j,m-1}^{(i)}$ such that
$$\zeta_j^{(i)} = \zeta_{j,0}^{(i)} + \zeta_{j,1}^{(i)}\omega + \cdots + \zeta_{j,m-1}^{(i)}\omega^{m-1} \text{ for } 1 \le i,\, j \le k.$$

```
    Initialisation: Computation of Structure Constants
(1) Use the multiplication table for G to compute
        • the number $k$ of conjugacy classes of $G$;
        • the exponent $m$ of $G$;
        • the sizes $h_1, h_2, \ldots, h_k$ of the classes $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_k$ (listed in
            some order);
        • the number $c_{r\,s\,t}$ of solutions $(x, y)$ of the equation $x \cdot y = z$
            for fixed $z \in \mathcal{C}_t$, such that $x \in \mathcal{C}_r$ and $y \in \mathcal{C}_s$, for $1 \le r,\, s,\, t \le k$;
        • integers $\langle\, i,\, s\,\rangle$ such that $g^s \in \mathcal{C}_{\langle\, i,\, s\,\rangle}$ if $g \in \mathcal{C}_i$, for
            $-1 \le s \le m-1$ and $1 \le i \le k$; set $i^* = \langle\, i,\, -1\,\rangle$.
    Construction of the Fields $\mathbb{Q}[\omega]$ and $\mathbb{F}_p$
(2) Compute the minimal polynomial $\Psi_m$ and an isolating region for an
    $m^{\text{th}}$ primitive root of unity, $\omega$, using the formulas
    $\omega = \cos\left(\frac{2\pi}{m}\right) + \sqrt{-1} \cdot \sin\left(\frac{2\pi}{m}\right)$   and   $\Psi_m = \prod_{d \text{ divides } m} (t^d - 1)^{\mu(m/d)}$.
(3) Set $p$ to be the smallest prime such that $p > 2n$ and $p \equiv 1 \pmod{m}$.
    Set $F = \mathbb{F}_p$.
(4) Compute and factor $\Psi_m$ in $F[t]$. Set $\hat{\omega}$ to be any root of $\Psi_m$ in $F[t]$.
    (Note that $\Psi_m$ splits into linear factors in this ring.)
```

Computation of a character table over F

(5) For $1 \leq s \leq k$, let $\hat{V}_s \in M_{k\times k}(F)$ such that $(\hat{V}_s)_{r\,t} = (c_{r\,s\,t} \bmod p)$ for $1 \leq r,\, s \leq k$. Compute the eigenvalues, and bases for the eigenspaces, of the matrices $\hat{V}_1, \hat{V}_2, \ldots, \hat{V}_k$. Compute bases for intersections of these eigenspaces (using a divide and conquer approach) to obtain vectors

$$\hat{w}_1 = \begin{bmatrix} \hat{\omega}_1^{(1)} \\ \hat{\omega}_2^{(1)} \\ \vdots \\ \hat{\omega}_k^{(1)} \end{bmatrix}, \quad \hat{w}_2 = \begin{bmatrix} \hat{\omega}_1^{(2)} \\ \hat{\omega}_2^{(2)} \\ \vdots \\ \hat{\omega}_k^{(2)} \end{bmatrix}, \quad \cdots \quad \hat{w}_k = \begin{bmatrix} \hat{\omega}_1^{(k)} \\ \hat{\omega}_2^{(k)} \\ \vdots \\ \hat{\omega}_k^{(k)} \end{bmatrix}$$

in $M_{k\times 1}(F)$, such that these are common eigenvectors of $\hat{V}_1, \hat{V}_2, \ldots, \hat{V}_k$, span $M_{k\times 1}(F)$, and such that $\hat{\omega}_1^{(i)} = 1$ for $1 \leq i \leq k$.

(6) For $1 \leq i \leq k$, compute $\hat{S}_i = h_i^{-1} \sum_{l=1}^{k} \hat{\omega}_l^{(i)} \hat{\omega}_{l^*}^{(i)}$. Set $\tilde{z}_i$ to be the unique element of $\mathbb{F}_p = F$ such that $\tilde{z}_i \in \{1, 2, \ldots, n\}$ and $\hat{S}_i \cdot \tilde{z}_i^2 = n$.

(7) For $1 \leq i,\, j \leq k$, compute

$$\hat{\zeta}_j^{(i)} = \frac{\tilde{z}_i \cdot \hat{\omega}_j^{(i)}}{h_j} \in F.$$

Recovery of a character table for G over $\mathbb{C}$

(8) For $1 \leq i,\, j \leq k$ and $0 \leq s \leq m$, compute

$$\hat{\zeta}_{j,s}^{(i)} = m^{-1} \cdot \sum_{t=0}^{m-1} \hat{\zeta}_{\langle j,\,t\rangle}^{(i)} \hat{\omega}^{-st};$$

$\hat{\zeta}_{j,s}^{(i)}$ is an element of $\mathbb{F}_p$ in the range $\{0, 1, \ldots, n\}$. Set $\zeta_{j,s}^{(i)} \in \mathbb{Z}$ to be the (unique) integer between 0 and n such that $\hat{\zeta}_{j,s}^{(i)} = (\zeta_{j,s}^{(i)} \bmod p)$.

(9) For $1 \leq i,\, j \leq k$, compute the $(i,\, j)^{\text{th}}$ entry of the character table for G over C,

$$\zeta_j^{(i)} = \sum_{s=0}^{m-1} \zeta_{j,s}^{(i)} \omega^s.$$

Return the desired values.

In fact, small primes do exist in these progressions. In 1930, Titchmarsch showed that the extended Riemann hypothesis implies the existence of a prime $p < k^{2+\epsilon}$ congruent to l modulo k, for any $\epsilon > 0$, $l \in \mathbb{Z}$, and for sufficiently large k (see [115], [23]). Linnik proved the existence of a prime $p < k^c$ congruent to l modulo k, for some constant $c > 0$ ("Linnik's constant"), for all $l \in \mathbb{Z}$ relatively prime with k, and (again) for sufficiently large k (see [83], [84]), without recourse to any unproved number theoretic assumptions. Subsequent research has included attempts to give concrete upper bounds for Linnik's constant; it has been shown recently that $c \leq 17$ ([20]; see also [60]). For an attempt to give a *lower* bound on the maximum size of a least prime in an arithmetic progression, see [97]; for more information about the distribution of primes in progressions and intervals, see Davenport [32].

These results imply that Dixon's algorithm has polynomial running time, if we choose the prime p by checking the leading terms in the arithmetic progression corresponding to the order n and exponent m of the input group G (as indicated above). We also have a guarantee that the algorithm is "practical" — that the algorithm has running time in $O(N^c)$ for input size N and for a *small* constant $c > 0$ — assuming the extended Riemann hypothesis. The results which do not depend on this hypothesis are less convincing; however, they are sufficient to show that this is a polynomial time algorithm.

Note also that, since the finite field $\mathbb{F}_p$ constructed by this algorithm has size (number of elements) which is *polynomial* in n, we can factor polynomials in $\mathbb{F}_p[t]$, and perform the other computations required by this algorithm, using Boolean circuits with polynomial size and with depth polynomial in the *logarithm* of the input size. In fact, we can implement the entire algorithm — including the construction of the finite field $\mathbb{F}_p$ — using Boolean circuits of this size, provided that we use a prime p which is slightly larger than that indicated in step 3 of the algorithm: If we choose p to be the smallest prime congruent to 1 modulo n (the order, rather than the exponent, of the group) which is larger than $2n$, then p can be treated as a "hardwired constant". That is, p is then a constant, dependent only on n, which does not have to be computed during the execution of the algorithm, but can be precomputed instead. (Now the fact that p is bounded by a polynomial function of n is of use in showing that our circuits can be *constructed* efficiently — and that we have a *log space uniform* family of Boolean circuits of small depth and size for character tables over $\mathbb{C}$ — rather than in showing that the resulting circuits are small.)

Thus, Dixon's algorithm can be used to prove the following fact.

Theorem 3.3.27. Given a multiplication table for a finite group G of order n, a character table for G over $\mathbb{C}$ can be computed using a uniform family of Boolean circuits of depth polynomial in log n and of size polynomial in n. That is, for computations over $\mathbb{C}$, "Computation of a Character Table" $\in$ NC.

3.3.4. A New Algorithm for Character Tables

In this section we suggest two improvements of Dixon's algorithm. We incorporate these to obtain a new probabilistic algorithm for the computation of character tables.

We first reduce the use of factorisation of polynomials, at the cost of introducing the possibility of failure. Recall that we compute vectors $w_1, w_2, \ldots, w_k$, and obtain the values $\omega_j^{(i)}$, for $1 \le i, j \le k$, by computing eigenvalues and bases for eigenspaces for a set of matrices, $V_1, V_2, \ldots, V_k$. In Example 3.3.25 we reduced the amount of work to be done by noting that one of the matrices (V_3) had a squarefree characteristic polynomial, and hence could be used alone to compute these values. We are not this lucky in general; there is no guarantee that some V_i will have the above property. However, we note that the vectors $w_1, w_2, \ldots, w_k$ (containing the $\omega_j^{(i)}$'s as entries) are eigenvalues of any matrix

$$V = c_1V_1 + c_2V_2 + \cdots + c_kV_k \qquad \text{for } c_1, c_2, \ldots, c_k \in F.$$

We will present a third, probabilistic version of Burnside's algorithm which makes a random choice of these constants $c_1, c_2, \ldots, c_k$ from F. If the resulting matrix V has a squarefree characteristic polynomial, then the vectors $w_1, w_2, \ldots, w_k$ can be computed by considering V alone. Otherwise, the algorithm fails* — though, with arbitrarily small positive probability, as indicated below.

Lemma 3.3.28. Let G be a finite group with n elements, and k conjugacy classes. Let F be a splitting field for G whose characteristic does not divide n, and let $\hat{F}$ be a subfield of F containing all entries of the character table for G over F. Let $V_1, V_2, \ldots, V_k \in M_{k\times k}(F)$ be the matrices used in Burnside's algorithm to generate a character table for G over F.

(i) If $|\hat{F}| \ge k$ then there exist constants $c_1, c_2, \ldots, c_k \in \hat{F}$ such that the matrix $V = c_1V_1 + c_2V_2 + \cdots + c_kV_k$ has k distinct eigenvalues.

(ii) Let $d \ge 1$ and suppose $|\hat{F}| \ge dk(2k-1)$. Then if $c_1, c_2, \ldots, c_k$ are chosen randomly and independently from a finite subset of $\hat{F}$ whose size is at least $dk(2k-1)$, then the probability that the resulting matrix V does not have k distinct eigenvalues is at most $1/d$.

Proof. We have noted that

$$\omega_s^{(i)} \cdot w_i = V_s \cdot w_i \qquad \text{for } 1 \le i,\ s \le k,$$

* This is clearly unnecessary, since Dixon's algorithm can be used if the characteristic polynomial of V is not squarefree; however, it simplifies presentation of the algorithm.

so that, for $c_1, c_2, \ldots, c_k \in F$,

$$(c_1\omega_1^{(i)} + c_2\omega_2^{(i)} + \cdots + c_k\omega_k^{(i)}) \cdot w_i = (c_1V_1 + c_2V_2 + \cdots + c_kV_k) \cdot w_i$$

for $1 \le i \le k$; the eigenvalues of $c_1V_1 + c_2V_2 + \cdots c_kV_k$ are the entries of the vector

$$w = \begin{bmatrix} \omega_1^{(1)} & \omega_2^{(1)} & \cdots & \omega_k^{(1)} \\ \omega_1^{(2)} & \omega_2^{(2)} & \cdots & \omega_k^{(2)} \\ \vdots & \vdots & \ddots & \vdots \\ \omega_1^{(k)} & \omega_2^{(k)} & \cdots & \omega_k^{(k)} \end{bmatrix} \cdot \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_k \end{bmatrix}.$$

We noted in proving Lemma 3.3.20 that the $k \times k$ matrix W in the above expression is nonsingular; since $\hat{F}$ contains all entries of a character table for G, it also includes all the entries $\omega_j^{(i)}$ of this matrix. That is, $W \in GL(k, \hat{F})$. It follows that we can set w to be any vector in $M_{k\times 1}(\hat{F})$, by using an appropriate choice of the constants $c_1, c_2, \ldots, c_k$. In particular, if $|\hat{F}| \ge k$, we can choose these constants so that the entries of the corresponding vector w (and the eigenvalues of V) are distinct — proving part (i) of the lemma.

To prove (ii), we note that if $V = y_1V_1 + y_2V_2 + \cdots + y_kV_k$, for indeterminates $y_1, y_2, \ldots, y_k$ over $\hat{F}$, then the characteristic polynomial ϕ of the matrix V has coefficients (in a new indeterminate, t) with total degree at most k in $y_1, y_2, \ldots, y_k$. The resultant, $\psi = \mathrm{res}_t(\phi, \frac{\mathrm{d}}{\mathrm{d}t}\phi)$, has total degree at most $k(2k-1)$ in these indeterminates, is not identically zero if $|\hat{F}| \ge dk(2k-1) \ge k$ (by part (i)), and is nonzero when evaluated at $(c_1, c_2, \ldots, c_k) \in \hat{F}^k$ if and only if the matrix $c_1V_1 + c_2V_2 + \cdots + c_kV_k$ has k distinct eigenvalues. We now apply the result of Schwartz (which we state as Proposition 2.4.23) to the polynomial ψ; part (ii) of the lemma follows immediately. ■

We also give a second modification of Dixon's algorithm, which allows us to prove an upper bound for running time of $O(N^c)$, for input size N and for a *small* constant c, without recourse to the extended Riemann hypothesis (see the end of Section 3.3.3 for a discussion of this hypothesis and its use in analysing Dixon's algorithm). In contrast with Dixon's algorithm, this algorithm performs computations over a "general" finite field $\mathbb{F}_q = \mathbb{F}_{p^k}$, for $p > 2n$, such that $\mathbb{F}_{p^k}$ includes an m^{th} primitive root of unity. We no longer require that $\mathbb{F}_q$ be a prime field — and we are no longer required to choose the characteristic p from an arithmetic progression.

We next consider the problem of finding a suitable finite field $\mathbb{F}_{p^k}$. We begin with the problem of choosing the characteristic, p. As noted above, we want p to be greater than $2n$; however, we don't want p to be much larger than this — it should be the case that computations over $\mathbb{F}_p$ are inexpensive. The next lemma indicates that a suitable prime can be chosen inexpensively — using time (asymptotically) less than $n^{1+\epsilon}$ for any positive ϵ (using the methods of Adleman, Pomerance, and Rumley [2] for certifying primes) if a deterministic method is required, or polynomial in $\log n$, using probabilistic methods (with a small, positive probability of failure).

We use the standard notation of $\pi(i)$ for the number of primes less than or equal to the positive integer i.

Lemma 3.3.29. Suppose $i \in \mathbb{N}$ and $i \geq 17$.

(i) $\pi(5i) - \pi(i) > \dfrac{i}{\ln\, i}$.

(ii) Let $\delta \in \mathbb{R}$ such that $\delta > 0$, and let $m = \lceil 2(\ln\, i)(\ln\, (1/\delta)) \rceil$. If we choose m odd integers randomly and independently from the set of odd integers greater than i and less than or equal to $5i$, then the probability that we fail to choose at least one prime is less than δ.

Proof. We use the inequality

$$\frac{i}{\ln\, i} < \pi(i) < 1.25506 \cdot \frac{i}{\ln\, i} \qquad \text{if } i \geq 17$$

stated by Rosser and Schoenfeld [105] (see their equations 3.5 and 3.6). Applying this, we have

$$\begin{aligned}
\pi(5i) - \pi(i) &> \frac{5i}{\ln\, (5i)} - 1.25506 \cdot \frac{i}{\ln\, i} \\
&= \frac{5i}{\ln\, i + \ln\, 5} - 1.25506 \cdot \frac{i}{\ln\, i} \\
&> \frac{5i}{2 \ln\, i} - 1.25506 \cdot \frac{i}{\ln\, i} \qquad \text{(since } i > 5) \\
&= (2.5 - 1.25506) \cdot \frac{i}{\ln\, i} \\
&> \frac{i}{\ln\, i},
\end{aligned}$$

as required to prove part (i).

To prove (ii), we note that there are $2i$ odd integers between i and $5i$; thus, if we choose a single odd integer from this set, then (by (i)) the probability that it is prime is at least $1/(2\ln i)$. If we choose m odd integers randomly and independently from this set, then the probability that all are composite is at most

$$\begin{aligned}\left(1-\frac{1}{2\ln\, i}\right)^m &= \left(1-\frac{\left(\frac{m}{2\ln\, m}\right)}{m}\right)^m \\ &< e^{-(m/(2\ln\, i))} \\ &\leq e^{-(2(\ln\, i)(\ln\,(1/\delta)))/(2\ln\, i)} \\ &= e^{-\ln\,(1/\delta)} \\ &= \delta, \qquad \text{as required to prove part (ii).} \quad \blacksquare\end{aligned}$$

We state our new algorithm on the following pages. The algorithm accepts a multiplication table for a finite group G as input, as well as a positive error tolerance, ϵ. The algorithm then either computes a character table for G over $\mathbb{C}$, or reports *failure* — failing with probability less than ϵ.

The probabilistic algorithm replaces the factorisation of k polynomials of degree k in $\mathbb{Q}[\omega]$ by factorisation of two polynomials over finite fields (one of degree at most m, over $\mathbb{F}_p$, and another of degree k over $\mathbb{F}_p[\hat{\omega}]$) and by some extra linear algebra — reducing the time used for the computation.

We can obtain a fourth, deterministic version of the algorithm by replacing the probabilistic search for the prime p by an exhaustive search — and by using the matrices $\hat{V}_1, \hat{V}_2, \ldots, \hat{V}_k$ to compute the vectors $\hat{w}_1, \hat{w}_2, \ldots, \hat{w}_k$ (as in Burnside's algorithm), instead of a single matrix, $\hat{V}$. While we increase the sequential time, and the size of Boolean circuits required to compute character tables over $\mathbb{C}$, these remain polynomial in n; the depths of Boolean circuits required for this computation (using either the probabilistic version, or the new deterministic one) are polynomial in $\log n$.

With the exception of step 1, Burnside's algorithm and the new probabilistic algorithm both use time polynomial in the *output size*, $k^2 m \log n$. Thus they efficiently solve the problem of computing a character table from the *structure constants* of a group and the additional values m, k, $h_1, h_2, \ldots, h_k$, and $\langle\, i,\, s\,\rangle$ (for $1 \leq i \leq k$ and $0 \leq s \leq m-1$). These constants have a (Boolean) representation of length $O(k^2 m \log n)$, matching the size of the character table to be generated; hence this size may be a more realistic measure of the "size" of a group (for this problem) than the number n of elements in G (provided, of course, that a set of structure constants for G is available).

Algorithm **Character Table III**

Input.
- Integer $n > 0$.
- A multiplication table for a finite group $G = \{ g_1, g_2, \ldots, g_n \}$ with n elements.
- An error tolerance $\epsilon > 0$.

Output. EITHER:
- The number k of conjugacy classes of G. (Note that k is also the number of distinct irreducible characters of G over $\mathbb{C}$.)
- An integer $m > 0$, such that the entries of a character table for G over $\mathbb{C}$ all lie in $\mathbb{Q}[\omega]$, for ω an m^{th} primitive root of unity.
- The minimal polynomial over $\mathbb{Q}$ and an isolating region in $\mathbb{C}$ for an m^{th} primitive root of unity, ω.
- A character table for G over $\mathbb{C}$, with k rows and columns, and with entry $\zeta_j^{(i)}$ given by integers $\zeta_{j,0}^{(i)}, \zeta_{j,1}^{(i)}, \ldots, \zeta_{j,m-1}^{(i)}$ such that

$$\zeta_j^{(i)} = \zeta_{j,0}^{(i)} + \zeta_{j,1}^{(i)}\omega + \cdots + \zeta_{k,m-1}^{(i)}\omega^{m-1}, \text{ for } 1 \le i,\ j \le k.$$

OR: *failure,* with probability less than ϵ.

Initialisation: Computation of Structure Constants.

(1) Use the multiplication table for G to compute
- the number k of conjugacy classes of G;
- the exponent m of G;
- the sizes $h_1, h_2, \ldots, h_k$ of the classes $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_k$ (listed in some order);
- the number $c_{r\,s\,t}$ of solutions (x, y) of the equation $x \cdot y = z$ for fixed $z \in \mathcal{C}_t$, such that $x \in \mathcal{C}_r$ and $y \in \mathcal{C}_s$, for $1 \le r,\ s,\ t \le k$;
- integers $\langle\, i,\ s \,\rangle$ such that $g^s \in \mathcal{C}_{\langle\, i,\, s,\, \rangle}$ if $g \in \mathcal{C}_i$, for $-1 \le s \le m - 1$ and $1 \le i \le k$; set $i^* = \langle\, i,\ -1 \,\rangle$.

Construction of the fields $\mathbb{Q}[\omega]$ *and* $\mathbb{F}_p[\hat{\omega}]$

(2) Compute the minimal polynomial Ψ_m and an isolating region for an m^{th} primitive root of unity, ω, using the formulas

$$\omega = \cos\left(\frac{2\pi}{m}\right) + \sqrt{-1} \cdot \sin\left(\frac{2\pi}{m}\right) \quad \text{and} \quad \Psi_m = \prod_{d \text{ divides } m} (t^d - 1)^{\mu(m/d)}.$$

(3) Set $N = \max(2n, \lceil 4k(2k-1)\epsilon^{-1} \rceil)$ and set $M = \lceil 2(\log_2 N)(\log_2(4\epsilon^{-1})) \rceil$. Choose M odd integers randomly and independently from the set $\{\, i \ :\ N < i \le 5N \,\}$.

(4) Use a probabilistic method to attempt to certify each integer as prime (so that any prime in this set is certified with probability at least $1 - (\epsilon/4M)$). If none of the M integers chosen in step 3 are certified as prime, then report *failure*. Otherwise, set p to be the smallest of these integers certified as prime, and go to step 5.

(5) Compute and factor Ψ_m in $\mathbb{F}_p[t]$. Set $f \in \mathbb{F}_p[t]$ to be an irreducible factor of Ψ_m. In following steps we will perform computations over $F = \mathbb{F}_p[\hat{\omega}] = F_p[t]/(f)$, for $\hat{\omega} = (t \bmod f)$.

Computation of a character table over F

(6) For $1 \leq s \leq k$, let $\hat{V}_s \in M_{k\times k}(F)$ such that $(\hat{V}_s)_{r\,t} = (c_{r\,s\,t} \bmod p)$ for $1 \leq r,\, s \leq k$. Let I be a subset of F with size $\lceil\, 4k(2k-1)\epsilon^{-1}\,\rceil$. Choose values $c_1,\, c_2,\, \ldots,\, c_k$ randomly and independently from I, and set $\hat{V} = c_1\hat{V}_1 + c_2\hat{V}_2 + \cdots + c_k\hat{V}_k$. Compute the characteristic polynomial of $\hat{V}$. If this polynomial is not squarefree in $F[t]$, then report *failure*. Otherwise, go to step 7.

(7) Since the eigenvalues of $\hat{V}$ are elements of $\mathbb{F}_p[\hat{\omega}] = F$, and the characteristic polynomial of $\hat{V}$ is squarefree, V has k distinct eigenvalues in F. Compute vectors

$$\hat{w}_1 = \begin{bmatrix} \hat{\omega}_1^{(1)} \\ \hat{\omega}_2^{(1)} \\ \vdots \\ \hat{\omega}_k^{(1)} \end{bmatrix}, \quad \hat{w}_2 = \begin{bmatrix} \hat{\omega}_1^{(2)} \\ \hat{\omega}_2^{(2)} \\ \vdots \\ \hat{\omega}_k^{(2)} \end{bmatrix}, \quad \cdots \quad \hat{w}_k = \begin{bmatrix} \hat{\omega}_1^{(k)} \\ \hat{\omega}_2^{(k)} \\ \vdots \\ \hat{\omega}_k^{(k)} \end{bmatrix}$$

in $M_{k\times 1}(F)$ so that these eigenvalues of $\hat{V}$, span $M_{k\times 1}(F)$, and so that $\hat{\omega}_1^{(i)} = 1$ for $1 \leq i \leq k$.

(8) For $1 \leq i \leq k$, compute $\hat{S}_i = h_i^{-1} \sum_{l=1}^{k} \hat{\omega}_l^{(i)} \hat{\omega}_{l^*}^{(i)}$. Set $\tilde{z}_i$ to be the unique element of $\mathbb{F}_p \subseteq F$ such that $\tilde{z}_i \in \{\,1,\, 2,\, \ldots,\, n\,\}$ and $\hat{S}_i \cdot \tilde{z}_i^2 = n$.

(9) For $1 \leq i,\, j \leq k$, compute

$$\hat{\zeta}_j^{(i)} = \frac{\tilde{z}_i \cdot \hat{\omega}_j^{(i)}}{h_j} \in F.$$

Recovery of a character table for G over $\mathbb{C}$

(10) For $1 \leq i,\, j \leq k$ and $0 \leq s < m$, compute

$$\hat{\zeta}_{j,s}^{(i)} = m^{-1} \cdot \sum_{t=0}^{m-1} \hat{\zeta}_{\langle\, j,\, t\,\rangle}^{(i)} \hat{\omega}^{-st};$$

$\hat{\zeta}_{j,s}^{(i)}$ is an element of $\mathbb{F}_p$ in the range $\{\,0,\, 1,\, \ldots,\, n\,\}$. Set $\zeta_{j,s}^{(i)} \in \mathbb{Z}$ to be the (unique) integer between 0 and n such that $\hat{\zeta}_{j,s}^{(i)} = (\zeta_{j,s}^{(i)} \bmod p)$.

(11) For $1 \leq i, j \leq k$, compute the $(i, j)^{\text{th}}$ entry of the character table for G over C,

$$\zeta_j^{(i)} = \sum_{s=0}^{m-1} \zeta_{j,s}^{(i)} \omega^s.$$

Return the desired values.

The preceding remarks about sequential time remain true if the exponent m of G is replaced in the algorithm by an integer $\hat{m} \leq m$, such that the entries of the character table for G over $\mathbb{C}$ all lie in $\mathbb{Q}[\omega]$, for ω an $\hat{m}^{\text{th}}$ primitive root of unity. The difference between m and $\hat{m}$ can be important: for example, the exponent of the symmetric group S_n is lcm(2, 3, ..., n), but it can be shown that the entries of the character table for S_n over $\mathbb{C}$ are all integers — so that we can set $\hat{m} = 1$. A set of structure constants for S_n, and a character table for S_n over $\mathbb{C}$, both have size polynomial in the number of conjugacy classes of S_n and hence in $2^{\sqrt{n}}$ (see Hua [64], Theorem 6.2), rather than in $|S_n|$, or in $2^{n \log n}$.

To conclude, we consider the question of whether the result "Computation of a Character Table over $\mathbb{C}$" $\in$ NC remains true when we consider computations using structure constants as inputs (and with the corresponding smaller input size). The answer depends on the efficiency of our choice of a prime $p > 2n$ — and, for the fourth, deterministic algorithm mentioned above, on the size of gaps between primes greater than $2n$. We will consider the probabilistic version of the new algorithm. It is clear that this requires circuits of size polynomial in $km \log(n\epsilon^{-1})$, and of depth polynomial in $\log(n \log \epsilon^{-1})$. (Note that we require certification of $O(\log(n \log \epsilon^{-1}))$-bit primes.) This motivates the following question.

Question 3.3.30. Does there exist a constant $c > 0$ such that, for every finite group G, if n is the number of elements of G and k is the number of conjugacy classes of G, then

$$\log_2 n \leq (\log_2 k)^c\,?$$

It is clear from inspection of the symmetric groups that if such a constant c exists, then $c > 2$. If such a constant exists, then our results imply that the problem "Computation of a Character Table over $\mathbb{C}$ from Structure Constants" is in the complexity class RNC — our probabilistic version of Burnside's algorithm can be implemented using Boolean circuits of depth polynomial in the logarithm of the input size, and of size polynomial in the input size. (Note also that it is *not* clear that this is also the case for Dixon's algorithm.) If no such constant c exists, then we must look for more efficient (parallel) methods for the certification of primes if we are to obtain this result using a version of Burnside's algorithm.

We also note a different variation on Dixon's algorithm, suggested by Schneider [108], who considers the problem of computing a character table for G given a more concise representation of the group than its multiplication table: Schneider considers the cost of computing a character table from a set of generators of a permutation group, and of computing a character table for a finitely presented p-group. In these cases, it is apparent that the cost of computing a set of structure constants for the group dominates the cost of computing the character table; Schneider's variation allows the character table to be computed without constructing a full set of structure constants in many cases. Schneider reports a significant decrease in the running time required to compute character tables for an impressive set of examples; further details can be found in [108].

3.3.5. Matrix Representations of Finite Groups

We now consider the problem of decomposing a matrix representation of a finite group, given both the image of each element of the group and a multiplication table for the group. Henceforth, we assume that $G = \{ g_1, g_2, \ldots, g_n \}$ is a finite group with n elements and with exponent m, and that a number field F is a splitting field for F (of characteristic zero), which includes an m^{th} primitive root of unity, ω.

We have already noted that it is trivial to decide whether a representation of a finite group over a field of characteristic zero is a completely reducible representation (see Proposition 3.3.11). It is also easy to decide whether two matrix representations of the same finite group are equivalent: It is a consequence of Proposition 3.3.18 that the irreducible characters of G over F are linearly independent functions over F. Thus, two matrix representations $\hat{T}_1$ and $\hat{T}_2$ are equivalent if and only if the trace of the matrices $\hat{T}_1(g)$ and $\hat{T}_2(g)$ are equal, for all $g \in G$. This condition is easily checked, given these matrices: We can solve this problem using Boolean circuits of size polynomial, and depth polylogarithmic, in the input size.

We next consider the problem of computing the isotypic components of a matrix representation $\hat{T} : G \to GL(k, F)$ (given the inputs described above). By Theorem 3.3.27, we can compute a character table for G over F efficiently in parallel. Suppose now that $\chi : G \to F$ is an irreducible character with degree d (so that $\chi(1_G) = d$); then the matrix

$$p_\chi = \frac{d}{n} \sum_{g \in G} \chi(g)^* \hat{T}(g)$$

is a projection onto the "carrier space" (a subspace of $M_{k\times 1}(F)$) for the isotypic component of $\hat{T}$ corresponding to the character χ. (Here, x^* denotes the complex conjugate of $x \in F$. Note that, since $\chi(g) \in \mathbb{Q}[\omega]$, $\chi(g)^* \in \mathbb{Q}[\omega] \subseteq F$ as well.) For a proof of this, see Serre [112] (Chapter 2, Theorem 8).

Now the matrix p_χ can be computed efficiently (in parallel), for each irreducible character χ, from a character table for G over F. Given each projection p_χ, the isotypic components of $\hat{T}$ are easily computed as well — proving the following.

Theorem 3.3.31. Let $G = \{ g_1, g_2, \ldots, g_n \}$ be a finite group with n elements and exponent m, and let F be a number field which includes an m^{th} primitive root of unity. Then, if we are given a multiplication table for G and the matrices $\hat{T}(g_i)$ for $1 \leq i \leq n$, for a matrix representation $\hat{T}$ of G over F, then the isotypic components of $\hat{T}$ can be computed using a uniform family of Boolean circuits with size polynomial, and depth polylogarithmic, in the input size.

We have seen that irreducible components of an isotypic matrix representation for G over $\mathbb{C}$ (or over F) can be computed, over a small extension of F, in probabilistic polynomial time (see Corollary 3.2.11). Babai and Rónyai have made a substantial improvement of this.

Theorem 3.3.32. (Babai and Rónyai [5]). Let G and F be as above. Then, if we are given a matrix representation $\hat{T}$ for G over F (by the matrices $\hat{T}(g_i)$ for $1 \leq i \leq n$), then a set of irreducible components for $\hat{T}$ over F can be computed (in F) by a deterministic Boolean algorithm, in polynomial time.

As Babai and Rónyai note, they also obtain a polynomial time algorithm for the computation of a character table for G over F which does *not* use Burnside's approach.

We conclude by noting an earlier, numerical algorithm for the decomposition of unitary matrix representations of finite groups: Dixon [35] presents an iterative method for the decomposition of a unitary matrix representation $U : G \to GL(k, \mathbb{C})$, given the matrices $U(g_1)$, $U(g_2)$, $\ldots$, $U(g_l)$ for a set of *generators* $g_1, g_2, \ldots, g_l$ of G. While Dixon shows that his method always (eventually) produces a decomposition, he does not discuss the asymptotic performance of his algorithm. Several questions remain to be answered: What is the order of convergence of the sequence Dixon uses to obtain a decomposition? What is the asymptotic running time of the algorithm? What is the effect of numerical error in the input on the performance of this method? We leave these questions for further work.

3.4. Computations for the Symmetric and Full Linear Groups

As in Section 3.3, we consider representations and characters for specific classes of groups. Again, we are chiefly interested in representations over fields of characteristic zero (and, in particular, over $\mathbb{R}$ and $\mathbb{C}$).

The literature concerning the representation theory of the symmetric groups and the linear groups is vast. In this section, we intend only to mention a few of the problems which have received attention, and to discuss some of the more commonly used algorithms for them. While we include a preliminary result here (Theorem 3.4.7), we consider the problems discussed here as subjects for further research. The reader interested in a more detailed presentation should refer to James and Kerber [69]. Hamermesh [61] and Wybourne [119] discuss applications of this theory to physical problems.

We consider computations for characters of the symmetric groups in Section 3.4.1; computations for the linear groups (and some of their subgroups) are considered in Section 3.4.2.

3.4.1. Computations for Characters of the Symmetric Groups

Since the symmetric group S_n of permutations of 1, 2, …, n is a finite group, with $n!$ elements, the results of Section 3.3 apply: we can compute character tables of S_n over $\mathbb{C}$ using polynomial time, or using Boolean circuits of size polynomial, and depth polylogarithmic, in the number of elements of S_n. However, we will see that the more interesting computational problems regarding characters of S_n (for physical applications) are more difficult — for most of this section, we consider problems which can be solved using a polynomial amount of *space,* but which are not known to be solvable in polynomial time.

A permutation $\pi \in S_n$ (viewed as a one-to-one, onto function from the set of integers $\{1, 2, \ldots, n\}$ to itself) can be specified completely by listing the image $\pi(i)$ of each integer i, for $1 \leq i \leq n$. For example, it is standard to write

$$\pi = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 5 & 3 & 4 & 2 & 1 \end{pmatrix}$$

to specify the permutation π such that $\pi(1) = 5$, $\pi(2) = 3$, $\pi(3) = 4$, $\pi(4) = 2$, and $\pi(5) = 1$; two rows are used, with the image $\pi(i)$ written below each integer i. Alternatively, we can specify the permutation π using *cycle notation* — with the points which are cyclically permuted grouped together in round brackets; using this notation, the above permutation can be written as

$$\pi = (2,\ 3,\ 4)(1,\ 5),$$

(with the image $\pi(i)$ following i, or with i at the end of the listing of a cycle and $\pi(i)$ at the beginning). The order in which the cycles appear in the listing, and the element beginning each cycle, can be arbitrarily chosen; we will list the cycles in decreasing order of length, and associate to a permutation π the list of integers $(l_1, l_2, \ldots, l_n)$ such that l_j is the length of the j^{th} cycle in this listing. We associate the list $(3, 2, 0, 0, 0)$ (or, leaving away the trailing zeros, $(3, 2)$) with the permutation π shown above.

It is clear that the list $\lambda = (l_1, l_2, \ldots, l_n)$ has the properties $l_1, l_2, \ldots, l_n \in \mathbb{Z}$, $l_1 \geq l_2 \geq \cdots \geq l_n \geq 0$, and $l_1 + l_2 + \cdots + l_n = n$. That is, λ is a *partition* of the integer n. Henceforth, we write this as "$\lambda \vdash n$".

It is easy to show that permutations π_1 and π_2 are conjugates in S_n if and only if π_1 and π_2 are associated with the same partition $\lambda = (l_1, l_2, \ldots, l_n)$. Conversely, for every partition of n there is a permutation π associated with λ. For example, we can set

$$\pi = (1, 2, \ldots, l_1)(l_1 + 1, l_1 + 2, \ldots, l_1 + l_2) \cdots (l_1 + l_2 + \cdots + l_{k-1} + 1, \ldots, n)$$

where k is the maximum integer such that $l_k \neq 0$.

Thus the number of conjugacy classes of S_n is the same as the number of partitions of n, and we have a natural association of each conjugacy class to a specific partition.

We estimate the number of entries in a character table for S_n by estimating (and squaring) $p(n)$, the number of partitions of n. The following estimates are proved (for example) in Hua [64] (see Section 8.6).

Proposition 3.4.1. If $n > 1$, then

$$2^{\lfloor \sqrt{n} \rfloor} < p(n) < n^{3\lfloor \sqrt{n} \rfloor}.$$

Proposition 3.4.2.

$$\lim_{n \to \infty} \frac{\log_e p(n)}{\sqrt{n}} = \pi \cdot \sqrt{\frac{2}{3}} \cong 2.56510;$$

thus, for any positive $\epsilon \in \mathbb{R}$,

$$p(n) < e^{(\pi \cdot \sqrt{(2/3)} + \epsilon)\sqrt{n}} = 2^{(\pi \cdot \sqrt{(2/3)} + \epsilon)(\log_2 e)\sqrt{n}} \qquad \text{for sufficiently large } n.$$

Thus a character table for S_n has fewer than $2^{2(\pi \cdot \sqrt{(2/3)} + \epsilon)(\log_2 e)\sqrt{n}}$ entries (for $\epsilon > 0$ and sufficiently large n). We will see later that these entries are all integers; since they are traces of matrices with order at most $n!$ and with roots of unity as characteristic values, each entry has absolute value at most $n!$, and has a Boolean representation with length $O(n \log n)$. One computational problem, then, is to construct a character table for S_n using a Boolean circuit with size as close as possible to the (Boolean) size of the table — that is, $O(2^{2(\pi \cdot \sqrt{(2/3)}+\epsilon)(\log_2 e)\sqrt{n}})$; the methods of Section 3.3 could be applied here.

Frobenius has provided an alternative method for computation of entries of this table. We now let F be an algebraically closed or real closed field of characteristic zero, and let $x_1, x_2, \ldots, x_n$ be indeterminates over F. To each partition $\lambda = (l_1, l_2, \ldots, l_n)$ of n we associate a symmetric polynomial s_λ:

$$s_\lambda = s_1^{\mu_1} s_2^{\mu_2} \cdots s_n^{\mu_n}, \tag{3.4.1}$$

where $s_r = (x_1^r + x_2^r + \cdots + x_n^r)$ for $1 \le r \le n$, and

$$\mu_r = |\ \{\, l_i : 1 \le i \le n \text{ and } l_i = r \,\}\ |.$$

For example, for $n = 7$ and $\lambda = (3, 3, 1)$,

$$s_\lambda = (x_1^3 + x_2^3 + x_3^3 + x_4^3 + x_5^3 + x_6^3 + x_7^3)^2 (x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7).$$

We set the polynomial $D(x_1, x_2, \ldots, x_n)$ to be the determinant of the Vandermonde matrix $V(x_1, x_2, \ldots, x_n) \in M_{n\times n}(F[x_1, x_2, \ldots, x_n])$, whose $(i, j)^{\text{th}}$ entry is x_j^{i-1} for $1 \le i, j \le n$; then

$$D(x_1, x_2, \ldots, x_n) = \prod_{i=1}^{n} \prod_{j=1}^{i-1} (x_i - x_j).$$

We associate to each partition $\lambda = (l_1, l_2, \ldots, l_n)$ a second polynomial,

$$\begin{aligned} t_\lambda &= \prod_{\pi \in S_n} (\operatorname{sgn}(\pi)) x_{\pi(1)}^{l_1+n-1} x_{\pi(2)}^{l_2+n-2} \cdots x_{\pi(n)}^{l_n} \\ &= \sum_{\pi \in S_n} \left[(\operatorname{sgn}(\pi)) \prod_{i=1}^{n} x_{\pi(i)}^{l_i+n-i} \right], \end{aligned} \tag{3.4.2}$$

for $\operatorname{sgn}(\pi)$ the *sign* of the permutation π (1 if π is an even permutation, -1 otherwise). For example, for $n = 4$ and $\lambda = (2, 1, 1)$,

$$\begin{aligned} t_\lambda &= \sum_{\pi \in S_4} \operatorname{sgn}(\pi) (x_{\pi(1)}^{2+4-1} x_{\pi(2)}^{1+4-2} x_{\pi(3)}^{1+4-3} x_{\pi(4)}^{0+4-4}) \\ &= \sum_{\pi \in S_4} \operatorname{sgn}(\pi) x_{\pi(1)}^5 x_{\pi(2)}^3 x_{\pi(3)}^2 \\ &= x_1^5x_2^3x_3^2 - x_1^5x_2^3x_4^2 - x_1^5x_3^3x_2^2 + x_1^5x_3^3x_4^2 + x_1^5x_4^3x_2^2 - x_1^5x_4^3x_3^2 - x_2^5x_1^3x_3^2 + x_2^5x_1^3x_4^2 \\ &\quad + x_2^5x_3^3x_1^2 - x_2^5x_3^3x_4^2 - x_2^5x_4^3x_1^2 + x_2^5x_4^3x_3^2 + x_3^5x_1^3x_2^2 - x_3^5x_1^3x_4^2 - x_3^5x_2^3x_1^2 + x_3^5x_2^3x_4^2 \\ &\quad + x_3^5x_4^3x_1^2 - x_3^5x_4^3x_2^2 - x_4^5x_1^3x_2^2 + x_4^5x_1^3x_3^2 + x_4^5x_2^3x_1^2 - x_4^5x_2^3x_3^2 - x_4^5x_3^3x_1^2 + x_4^5x_3^3x_2^2. \end{aligned}$$

The polynomials D and t_λ are clearly *antisymmetric:* we change the sign of the values of these polynomials by transposing the values of two indeterminates x_i and x_j (for $i \neq j$).

Note that for each term

$$x_{\pi(1)}^{l_1+n-1} x_{\pi(2)}^{l_2+n-2} \cdots x_{\pi(n)}^{l_n+n-n}$$

of each t_λ, the indeterminates $x_1, x_2, \ldots, x_n$ occur with distinct degrees — and that the degree sequence l_1+n-1, l_2+n-2, $\ldots$, l_n+n-n is strictly decreasing and determines the partition $\lambda = (l_1, l_2, \ldots, l_n)$ (since $l_1 \geq l_2 \geq \cdots \geq l_n$). Clearly, then, the polynomials t_λ (for all partitions λ of n) are linearly independent in $F[x_1, x_2, \ldots, x_n]$ — and the following formula can be used to compute entries of the character table of S_n.

Proposition 3.4.3. (Frobenius) Let F be an algebraically closed, or real closed, field of characteristic 0, and let $n \in \mathbb{Z}$ with $n > 0$. Then, for each partition λ of n, there exists an irreducible character $\zeta^{(\lambda)}$ of S_n, whose value $\zeta_\mu^{(\lambda)}$ at any permutation $\pi \in S_n$ in the conjugacy class corresponding to a partition μ of n, is determined by the expression

$$s_\mu D = \sum_{\nu \vdash n} \zeta_\mu^{(\nu)} t_\nu.$$

If λ_1 and λ_2 are distinct partitions of n, then the corresponding characters $\zeta^{(\lambda_1)}$ and $\zeta^{(\lambda_2)}$ are also distinct.

Proposition 3.4.3 is well known; a derivation of the above formula is given by Hamermesh [61] (Sections 7.1–7.2). It gives us a means of computing all the entries in a column of a character table for S_n, by computing the coefficients of the polynomial $s_\mu D$, and expressing $s_\mu D$ as a linear combination of the t_ν's. It also establishes that each entry $\zeta_\mu^{(\nu)} \in \mathbb{Q}$ — and, since $\zeta_\mu^{(\nu)}$ is known to be an algebraic integer, $\zeta_\mu^{(\nu)} \in \mathbb{Z}$. Finally, it allows us to specify a particular irreducible character ζ^ν of S_n by listing the corresponding partition ν, and to pose the following problem.

Problem	**Entry of a Character Table for S_n**
Input.	• Integer $n > 0$. • Partitions λ, μ of n.
Output.	• The integer $\zeta_\mu^{(\lambda)}$.

We have not discussed a corresponding problem for arbitrary finite groups, even though the number of irreducible characters (over $\mathbb{C}$) of such a group, G, is known to be the number of conjugacy classes of G, because we generally have no useful way to indicate a specific character of G — or to define a problem "Entry of a Character Table for G".

Now our methods for computing character tables are still applicable (in principle). However, the input and output for our problem are no longer a multiplication table (or a set of structure constants) and a character table for S_n — with size $\Omega(2^{\sqrt{n}})$; instead, we use as input a pair of partitions of n, and compute a single entry of the character table — each with Boolean representations of length $O(n \log n)$, polynomial in the *logarithm* of the old input and output size. Consequently, while we showed in Section 3.3 that the computation of character tables (over $\mathbb{C}$) for finite groups could be performed using time polynomial and space polylogarithmic (or in parallel, using time polylogarithmic and a polynomial number of processors) in the input size, we will analyse a commonly used algorithm for "Entry of a Character Table for S_n" in order to show that the problem can be solved using polynomial *space*. The question of whether it can be solved using polynomial time remains open.

The method we consider uses the formula of Frobenius, stated in Proposition 3.4.3, to compute the integer $\zeta_\mu^{(\nu)}$. Suppose $\mu = (m_1, m_2, \ldots, m_k, m_{k+1}, \ldots, m_n)$, with $m_k > m_{k+1} = m_{k+2} + \cdots + m_n = 0$. Then

$$s_\mu = \prod_{i=1}^{k} (x_1^{m_i} + x_2^{m_i} + \cdots + x_n^{m_i}),$$

and the algorithm proceeds by computing the polynomial

$$\left[\prod_{i=1}^{l} (x_1^{m_i} + x_2^{m_i} + \cdots + x_n^{m_i}) \right] D$$

for $l = 1, 2, \ldots, k$, and expressing each product as a linear combination of polynomials "t_ϕ" (for ϕ a partition of an integer less than or equal to n* — specifically, for $m = m_1 + m_2 + \cdots + m_l$ at the l^{th} stage of the computation).

The method is generally presented as a "graphical" or "combinatorial" method, with conjugacy classes of S_n, irreducible characters of S_n, partitions of n, and the polynomials t_ϕ all represented by *Young diagrams* for the associated partitions: the partition $\lambda = (l_1, l_2, \ldots, l_n)$ is represented by a diagram of dots (or ×'s, or boxes)

* Here we use as t_ϕ, for ϕ a partition of $m \leq n$, a polynomial with total degree $(n(n-1))/2+m$ in the indeterminates $x_1, x_2, \ldots, x_n$. If $\phi = (l_1, l_2, \ldots, l_m)$ then we define s_ϕ and t_ϕ as in Formulas 3.4.1 and 3.4.2, using $l_{m+1} = l_{m+2} = \cdots = l_n = 0$.

placed in m rows (for $m = \max\{\, i : l_i \neq 0\,\}$) and with l_i entries left-adjusted in row i. Using this representation, the product

$$(x_1^k + x_2^k + \cdots + x_n^k)t_\phi$$

is computed (and represented as a linear combination of t_ψ's for partitions ψ) by a "regular application of k dots" to the Young diagram representing t_ϕ. The method also includes a "pruning" operation which discards all intermediate Young diagrams (and corresponding polynomials t_ϕ) which cannot possibly contribute to the coefficient being computed. The method is described in detail (with a derivation from Frobenius' formula) by Hamermesh [61] (Section 7.4); Hamermesh also provides examples of its use. A straightforward analysis of the method, using the bounds on the number of partitions of n stated as Propositions 3.4.1 and 3.4.2, yields the bounds stated below.

Theorem 3.4.4. Given an integer $n > 0$ and partitions λ and μ of n, a binary representation of the integer $\zeta_\mu^{(\lambda)}$ can be computed using a (log space) uniform family of Boolean circuits of size $O(2^{(\pi\cdot\sqrt{(2/3)}+\epsilon)(\log_2 e)\sqrt{n}})$ and depth $n^{O(1)}$, for $\epsilon > 0$ and sufficiently large n.

Corollary 3.4.5. For n, λ, and μ as above, a binary representation of the integer $\zeta_\mu^{(\lambda)}$ can be computed by a sequential deterministic (Boolean) algorithm, using time and space $O(2^{(\pi\cdot\sqrt{(2/3)}+\epsilon)(\log_2 e)\sqrt{n}})$, for constant $\epsilon > 0$ and sufficiently large n.

Corollary 3.4.6. For n, λ, and μ as above, the integer $\zeta_\mu^{(\lambda)}$ can be computed by a sequential deterministic (Boolean) algorithm, using time $O(n^{n+c})$ and space $n^{O(1)}$, for a constant $c > 0$. Thus, "Entry of a Character Table for S_n" $\in$ PSPACE.

The bound stated in Corollary 3.4.6 is obtained by noting that at most n polynomials t_ϕ arise by the multiplication of a fixed t_λ by $(x_1^k + x_2^k + \cdots + x_n^k)$ at each stage of the algorithm (or, alternatively, that at most n Young diagrams are obtained from a single one by the "regular application of k dots") — and by noting that the algorithm has at most n stages.

It seems natural to ask how much we save by the "pruning" performed by the algorithm: Is the algorithm much faster than our (easily obtained) upper bounds indicate? We note that for some inputs, the combinatorial algorithm requires sequential time $\Omega(2^{c\sqrt{n}})$ for a constant $c > 0$.

Theorem 3.4.7. For any integer $n > 1$, let $\mu = (1, 1, \ldots, 1)$ be a partition of n, and let $\lambda = (l_1, l_2, \ldots, l_n)$ be the partition of n such that

- $l_i = \lfloor \sqrt{n} \rfloor$, for $1 \leq i \leq \lfloor \left(\frac{n}{\lfloor \sqrt{n} \rfloor} \right) \rfloor$;
- $l_i = n - \lfloor \left(\frac{n}{\lfloor \sqrt{n} \rfloor} \right) \rfloor \cdot \lfloor \sqrt{n} \rfloor$, for $i = 1 + \lfloor \left(\frac{n}{\lfloor \sqrt{n} \rfloor} \right) \rfloor$;
- $l_i = 0$ for $i > 1 + \lfloor \left(\frac{n}{\lfloor \sqrt{n} \rfloor} \right) \rfloor$.

Then the graphical algorithm discussed above must examine $\Omega\left(2^{2\lfloor \sqrt{n} \rfloor}/\sqrt{n}\right)$ partitions of integers less than or equal to n in order to compute $\zeta_\mu^{(\lambda)}$.

We have chosen λ and μ as above to guarantee that all the partitions $(l_1, l_2, \ldots, l_k)$ of integers less than n, such that $l_1 \leq \lfloor \sqrt{n} \rfloor$, and with at most $\lfloor \sqrt{n} \rfloor$ parts (that is, so that $l_i = 0$ for $i > \lfloor \sqrt{n} \rfloor$) must be examined — that is, so that each contributes a positive value to the coefficient being computed. Since there are as many of these partitions as there are nondecreasing functions from the set $\{1, 2, \ldots, \lfloor \sqrt{n} \rfloor\}$ to itself (in particular, $\Omega(2^{2\lfloor \sqrt{n} \rfloor}/\lfloor \sqrt{n} \rfloor)$ of them), the bound follows immediately.

We view this result as preliminary. Since several other algorithms for this computation exist, and some compute entries of particular rows or columns of the character table very quickly, Theorem 3.4.7 can only be viewed as evidence that a specific algorithm for this problem is asymptotically inefficient when applied to a specific (relatively small) set of instances of this problem. Results showing that all known algorithms perform badly for a nontrivial fraction of all inputs, or (better yet) showing that the problem is actually hard, would be more interesting. Unfortunately, we do are not aware of any such results.

We next define two problems concerning the decomposition of "products" of characters of S_n. While the corresponding problems for matrix representations are straightforward (and can be solved in expected polynomial time, by the methods of Sections 2 and 3), we are given only the characters for these representations as input for the problems to be discussed.

Suppose first that $T_1 : S_n \to GL(m_1, F)$ and $T_2 : S_n \to GL(m_2, F)$ are matrix representations of S_n. The *inner product* of T_1 and T_2 is the tensor product $T_1 \otimes_F T_2$, of dimension $m_1 m_2$ over F, which we discussed in Example 3.1.7. A natural problem (for matrix representations of S_n) would be to form, and decompose, the inner product of two given representations. Instead of posing this, we note that if ϕ_1 and ϕ_2 are two *characters* of S_n, then we can also generate a third character, $\phi_1 \times \phi_2$, as the character of the representation $T_1 \otimes_F T_2$ obtained from representations T_1 and T_2, with characters ϕ_1 and ϕ_2 respectively. The *inner product* $\phi_1 \times \phi_2$ of ϕ_1 and ϕ_2 will, in fact, be well defined: it will be independent of the choice of the representations T_1 and T_2 (each chosen from those representations with the desired character). In general, the problem of decomposing the inner product $\phi_1 \times \phi_2$ of two *irreducible* characters is considered. Again, we consider characters over algebraically closed or real closed fields of characteristic zero — and, in particular, over $\mathbb{C}$ and $\mathbb{R}$.

Problem **Decomposition of Inner Product**

Input.
- Integer $n > 0$.
- Partitions λ_1, λ_2, and μ of S_n.

Output.
- The coefficient c_μ of $\zeta^{(\mu)}$ in the decomposition $\zeta^{(\lambda_1)} \times \zeta^{(\lambda_2)} = \sum_{\phi \vdash n} c_\phi \zeta^{(\phi)}$.

Again, we have a problem with input of size $O(n \log n)$ which concerns characters of S_n. As noted in Section 3.3, the value of the character $\zeta^{(\lambda_1)} \times \zeta^{(\lambda_2)}$ at an element g of S_n is the product of $\zeta^{(\lambda_1)}(g)$ and $\zeta^{(\lambda_2)}(g)$, as our notation suggests. It is a consequence of the orthogonality relations for characters of finite groups (in particular, Proposition 3.3.18(ii)), that

$$c_\mu = n^{-1} \sum_{g \in S_n} \zeta^{(\lambda_1)}(g) \zeta^{(\lambda_2)}(g) \zeta^{(\mu)}(g^{-1}).$$

Applying the upper bounds of Section 3.4.1, we conclude that the problem "Decomposition of Inner Product" is in PSPACE. While we can use a slightly more efficient process to compute c_μ (namely, by computing the sizes of conjugacy classes of S_n and using Proposition 3.3.18(iii), so that we sum over partitions of n instead of over all elements of S_n), no efficient (polynomial-time) method for this computation is known.

We will describe, rather than formally define, an outer product of characters. Given matrix representations $T_1 : S_{n_1} \to GL(m_1, F)$ and $T_2 : S_{n_2} \to GL(m_2, F)$, there is a natural matrix representation $T_1 + T_2$ of $S_{n_1} \oplus S_{n_2}$, of dimension $m_1 + m_2$ over F: For each element g_1 of S_{n_1} and g_2 of S_{n_2}, we set

$$(T_1 + T_2)((g_1, g_2)) = \begin{bmatrix} T_1(g_1) & 0 \\ 0 & T_2(g_2) \end{bmatrix} \in GL(m_1 + m_2, F).$$

Now we view $S_{n_1} \oplus S_{n_2}$ as a subgroup of $S_{n_1+n_2}$ (namely, the set of permutations mapping the sets $\{ 1, 2, \ldots, n_1 \}$ and $\{ n_1 + 1, n_1 + 2, \ldots, n_1 + n_2 \}$ to themselves): given $(g_1, g_2) \in S_{n_1} \oplus S_{n_2}$, we use the element g_1 of S_{n_1} to define the action of (g_1, g_2) on $1, 2, \ldots, n_1$, and we use $g_2 \in S_{n_2}$ to define the action on $n_1 + 1, n_1 + 2, \ldots, n_1 + n_2$. The *outer product* of T_1 and T_2 is the matrix representation of $S_{n_1+n_2}$ *induced* by the representation $T_1 + T_2$ of this subgroup (isomorphic to $S_{n_1} \oplus S_{n_2}$) of $S_{n_1+n_2}$. (Representations of groups *induced* by representations of subgroups are discussed in most treatments of basic representation theory. In particular, Serre's treatment of this subject is short and readable; see [112], Section 3.3.)

Again, we have a corresponding, well defined, operation on characters of the symmetric group. We will denote by $\phi_1 \boxtimes \phi_2$ the character (of $S_{n_1+n_2}$) obtained as the outer product of characters ϕ_1 and ϕ_2 of S_{n_1} and S_{n_2}, respectively.*

Problem **Decomposition of Outer Product**

Input.
- Integers n_1, $n_2 > 0$.
- Partitions λ_1, λ_2, and μ of n_1, n_2, and $n_1 + n_2$ respectively.

Output.
- The coefficient c_μ of $\zeta^{(\mu)}$ in the decomposition
$$\zeta^{(\lambda_1)} \boxtimes \zeta^{(\lambda_2)} = \sum_{\phi \vdash n_1+n_2} c_\phi \zeta^{(\phi)}.$$

Again, there is a well known graphical method commonly used to solve this problem, (based on) the *Littlewood-Richardson rule.* The method was introduced by Littlewood and Richardson in 1934 ([86]); the first complete proof of correctness of the method has appeared more recently (See the monograph of Macdonald [88] for a combinatorial proof). The method is often presented in treatments of the representation theory of the symmetric group (for example, see James and Kerber [69], and Hamermesh [61]). Remmel and Whitney [101] have recently presented an alternative method, which they claim to be more efficient for a number of applications than the Littlewood-Richardson rule. An analysis of either method can be used to show that the problem "Decomposition of Outer Product" is in PSPACE; to our knowledge, no analysis of the worst case running time for either method has appeared.

* Our notation is nonstandard. Unfortunately, several symbols are in use to denote this operation (and to denote operations defined in Section 3.4.2) — and different authors use the same symbol to denote different operations. We choose our notation with the intention of avoiding symbols which have been used for other operations — and, in particular, to avoid the tensor symbol, ⊗, which is overused.

3.4.2. Computations for Characters of the Full Linear Groups

We now consider some problems concerning representations and characters of the full linear group, $GL(n, F)$, of nonsingular matrices of order n over a field F. Again, we are chiefly concerned with the case that F is an algebraically closed or real closed field of characteristic zero — and we will consider representations of $GL(n, F)$ over the (same) field F.

We first note that if $F = \mathbb{R}$ or $F = \mathbb{C}$ then not all representations of $GL(n, F)$ are completely reducible.

Example 3.4.8. Let $F = \mathbb{R}$ (or $F = \mathbb{C}$), $n > 1$, and let $T : GL(n, F) \to GL(2, F)$ such that, for $A \in GL(n, F)$,

$$T(A) = \begin{bmatrix} 1 & \log_2 |\det(A)| \\ 0 & 1 \end{bmatrix}.$$

Then T is a matrix representation of $GL(n, F)$ over F which is not completely reducible.

We restrict attention to a subset of the matrix representations of $GL(n, F)$, all of which are completely reducible (for F a field of characteristic 0) — namely, the *tensor representations* of $GL(n, F)$ over F. The irreducible representations in this set are the irreducible components of the matrix representations $T^{(i)}$, for $i \in \mathbb{N}$, where $T^{(0)} : GL(n, F) \to GL(1, F)$ (with $T^{(0)}(A) = [1]$ for all $A \in GL(n, F)$), $T^{(1)} : GL(n, F) \to GL(n, F)$ (with $T^{(1)}(A) = A$ for all $A \in GL(n, F)$), and with $T^{(i+1)} = T^{(i)} \otimes_F T^{(1)}$, the i^{th} tensor power of $T^{(1)}$, for $i > 0$. Thus $T^{(i)} : GL(n, F) \to GL(n^i, F)$.

The decomposition of these tensor powers is discussed by Hamermesh [61] (See Chapter 10). As is noted there, there are as many (inequivalent) irreducible components of $T^{(m)}$ as there are partitions of m with at most n nonzero parts (that is, partitions $(l_1, l_2, \ldots, l_n)$ with $l_1 \geq l_2 \geq \cdots \geq l_n \geq l_{n+1} = 0$ and $l_1 + l_2 + \cdots + l_n = m$). These irreducible components have distinct characters; there is a natural correspondence between these irreducible characters and partitions with at most n parts. For example, if $n = 3$ then there are inequivalent irreducible components of $T^{(5)}$ corresponding to the partitions

$$(5),\ (4, 1),\ (3, 1, 1),\ \text{and}\ (2, 2, 1).$$

We denote the corresponding characters by $\{5\}$, $\{4, 1\}$, $\{3, 1, 1\}$, and $\{2, 2, 1\}$, respectively.

Problem **Evaluation of a Tensor Character of $GL(n, F)$**

Input.
- Integers n, $m > 0$.
- A partition λ of m with at most n parts.
- A matrix $A \in GL(n, F)$.

Output.
- The value $\{\lambda\}(A)$ of the character $\{\lambda\}$ at the matrix A.

In fact, the characters $\{\lambda\}$ induce well defined functions from the characteristic values of the matrices in $GL(n, F)$ to F: the value $\{\lambda\}(A)$ is $e_\lambda(c_1, c_2, \ldots, c_n)$, where $e_\lambda : F^n \to F$ is the *Schur function* (of n indeterminates) corresponding to the partition λ of m, and $c_1, c_2, \ldots, c_n$ are the characteristic values of A. In general, if λ is a partition of m with at most n parts, then e_λ is a symmetric polynomial with total degree m. (If λ is as above, but $l_i \neq 0$ for $i > n$, then $e_\lambda = 0$.) The value $\{\lambda\}(A)$ depends only on the characteristic values of the matrix A (and the partition λ); in principle, we could compute this value by factoring the characteristic polynomial of A to generate the characteristic values $c_1, c_2, \ldots, c_n$, and then evaluate $e_\lambda(c_1, c_2, \ldots, c_n)$. In fact, we can do better than this. Since the characteristic polynomial of A is $\prod_{i=1}^{n}(t - c_i)$, the coefficient of t^k for this polynomial is

$$(-1)^{n-k} a_{n-k}(c_1, c_2, \ldots, c_n),$$

for a_{n-k} the elementary symmetric polynomial of degree $n - k$ in n indeterminates. For $0 \le i \le n$, the value $a_i(c_1, c_2, \ldots, c_n)$ can be computed directly from the entries of A (simply by computing A's characteristic polynomial). Now we apply the following identity, stated by Stanley [113] (as Corollary 11.2).

Proposition 3.4.9. Let $\lambda = (l_1, l_2, \ldots, l_n)$ be a partition with largest part $l_1 = q$. Then $e_\lambda(c_1, c_2, \ldots c_n)$ is the determinant of the matrix

$$\begin{bmatrix} a_{m_1}(c_1, c_2, \ldots, c_n) & a_{m_1+1}(c_1, c_2, \ldots, c_n) & \cdots & a_{m_1+q-1}(c_1, c_2, \ldots, c_n) \\ a_{m_2-1}(c_1, c_2, \ldots, c_n) & a_{m_2}(c_1, c_2, \ldots, c_n) & \cdots & a_{m_2+q-2}(c_1, c_2, \ldots, c_n) \\ \vdots & \vdots & \ddots & \vdots \\ a_{m_q-q+1}(c_1, c_2, \ldots, c_n) & a_{m_q-q+2}(c_1, c_2, \ldots, c_n) & \cdots & a_{m_q}(c_1, c_2, \ldots, c_n) \end{bmatrix},$$

where $a_i(c_1, c_2, \ldots, c_n) = 0$ if $i < 0$ and $\mu = (m_1, m_2, \ldots, m_q)$ is the *conjugate partition* of λ: that is, $m_i = |\{\, j : l_j \ge i \,\}|$, for $1 \le i \le q$.

It follows immediately that the problem "Evaluation of a Tensor Character of $\mathrm{GL}(n, F)$" can be solved using arithmetic-Boolean circuits over F, of size polynomial in nm and depth polylogarithmic in nm. It the entries of A are represented as elements of a number field, then the corresponding statement holds for computation by Boolean circuits. If we consider the integers n and m to be represented (as inputs for this problem) in unary, rather than binary notation, so that the input size matches the output size, then we can conclude that this version of the problem is in NC.

We next consider the problem of decomposing tensor products. We denote [†] by $\{\lambda_1\}\boxdot\{\lambda_2\}$ the "inner tensor product" of the characters $\{\lambda_1\}$ and $\{\lambda_2\}$ of $GL(n, F)$ — that is, the character of a representation $T_1 \otimes T_2$ of $GL(n, F)$ obtained from representations T_1 and T_2 of $GL(n, F)$ with characters $\{\lambda_1\}$ and $\{\lambda_2\}$ respectively.

Recall that if $\{\lambda_i\}$ is a partition of m_i with at most n parts, for $i = 1, 2$, then the Schur function e_{λ_i} corresponding to λ_i has total degree m_i. It is clear, then, that the symmetric function (of characteristic values) corresponding to the product $\{\lambda_1\}\boxdot\{\lambda_2\}$ has total degree $m_1 + m_2$, and that the irreducible components of the *character* $\{\lambda_1\}\boxdot\{\lambda_2\}$ will correspond to partitions of $m_1 + m_2$.

Problem **Decomposition of a Tensor Product of Representations of $GL(n, F)$**

Input.
- Integers n, m_1, $m_2 > 0$.
- Partitions λ_1 of m_1, λ_2 of m_2, and μ of $m_1 + m_2$, each with at most n parts.

Output.
- The coefficient c_μ of $\{\mu\}$ in the decomposition $\{\lambda_1\}\boxdot\{\lambda_2\} = \sum_{\mu\vdash m_1+m_2} c_\mu\{\mu\}$.

This problem is closely related to one which we have already discussed.

Proposition 3.4.10. Let F be an algebraically closed or real closed field of characteristic 0. Let λ_1 and λ_2 be partitions of integers m_1 and m_2 respectively, each with at most n parts. Then if integers c_μ are given (for $\mu \vdash m_1 + m_2$, with at most n parts) by the relation

$$\zeta^{(\lambda_1)}\boxtimes\zeta^{(\lambda_2)} = \sum_{\mu\vdash m_1+m_2} c_\mu \zeta^{(\mu)},$$

for characters $\zeta^{(\lambda_1)}$ and $\zeta^{(\lambda_2)}$ of S_{m_1} and S_{m_2} respectively, then

$$\{\lambda_1\}\boxdot\{\lambda_2\} = \sum_{\mu\vdash m_1+m_2} c_\mu\{\mu\},$$

for the same constants c_μ and for the characters $\{\lambda_1\}$ and $\{\lambda_2\}$ of $GL(n, F)$.

[†] Again, this notation is nonstandard.

We set $\{\mu\} = 0$ if μ is a partition with more than n nonzero parts.

This correspondence between problems is well known (see, for example, [119] or [78]). As a consequence of this, we can decompose a tensor product of irreducible characters of $GL(n, F)$ using methods for the decomposition of outer products of characters of the symmetric groups — in particular, using the Littlewood-Richardson rule and the method of Remmel and Whitney — and so we can conclude that the problem of decomposing tensor products of characters of $GL(n, F)$ is in PSPACE.

For a discussion of a further problem (from algebraic geometry) which is equivalent to the decomposition of outer products of representations of the symmetric groups in this way, refer to [78] (and, for a more general discussion of the background for the problem, to [73]).

Having discussed representations of $GL(n, F)$, we can introduce the notion of a *plethysm* of representations. Suppose $\phi : G \to GL(n, F)$ is a representation of a group G, and $\psi : GL(n, F) \to GL(m, F)$ is a representation of $GL(n, F)$. It is easily checked that the *plethysm* $\psi \circledcirc \phi$ of ψ and ϕ, defined[‡] by

$$\psi \circledcirc \phi(g) = \psi(\phi(g)) \qquad \text{for } g \in G,$$

is also a representation of G. Viewed as an operator on (matrix) representations, then *plethysm* is simply a composition of functions. As is the case for the other operators we have discussed in this section, "plethysm" is more often considered as a (well defined) operator on characters.[*]

Again, we consider the problems of evaluation and decomposition. If the dimension, n, of the representation ϕ is small (and, in particular, if it is feasible to compute the values $\phi(1)$, $\phi(g)$, $\ldots$, $\phi(g^{n-1})$ for an element g of G) then we can evaluate $\psi(\phi(g))$ using the fact that (if F has characteristic 0) ψ can be considered not only as a function of the characteristic values of the matrix $\hat{T}(g)$, for $\hat{T}$ a matrix representation with character ϕ, but also as a function of the "power sum" symmetric functions of these characteristic values — that is, as a function of the values $\phi(1)$, $\phi(g)$, $\phi(g^2)$, $\ldots$, $\phi(g^{n-1})$. Given $\phi(1)$, $\phi(g)$, $\ldots$, $\phi(g^{n-1})$, the value $\psi \circledcirc \phi$ can be computed using arithmetic-Boolean circuits of size polynomial in nm and depth

[‡] Once again, we are using a nonstandard symbol to replace $\otimes$

[*] In fact, our definition in terms of matrix representations is nonstandard. While our definition is occasionally mentioned as an equivalent, alternative definition (see [42], [116], and [119]), "plethysm" is more often defined as an operator for characters — or, using the relationship between characters of $GL(n, F)$ and symmetric polynomials noted above, as an operator on symmetric polynomials (see [21] and [88]). For a discussion of plethysm as an operator on *partitions*, and on characters of the symmetric group, see [69].

polylogarithmic in nm, for ψ corresponding to a partition of m, and for n the dimension of ϕ.

It is more frequently the case that we are asked to evaluate a plethysm $\phi \circledcirc \psi$, for ψ a character of the full linear group $GL(k, F)$ (or for a subgroup of $GL(k, F)$ such as the group of unitary, orthogonal, or symplectic $k \times k$ matrices over F). In these cases, the dimension (n) of ψ is generally too large for us to evaluate the plethysm by the above method. If ψ is an irreducible character of a full matrix group $GL(k, F)$ then we can evaluate $\psi \circledcirc \phi$ at a matrix $A \in GL(k, F)$ given either the entries of A, or the characteristic values of A, or the coefficients of the characteristic polynomial, using arithmetic-Boolean circuits of size polynomial in $m_1 m_2$ (where $\phi \vdash m_1$ and $\psi \vdash m_2$) by using standard identities for plethysms of Schur functions. (Note formulas 1.5(a) and 2.7 of Chen, Garsia, and Remmel [21].)

We next examine the problem of decomposing plethysms. If ψ is a character of a finite group then we can decompose a plethysm $\phi \circledcirc \psi$ by evaluating this at an element of each conjugacy class of G, and then using the orthogonality relations (Proposition 3.3.18) and a character table for G, to express the plethysm as a sum of irreducible characters. A similar, more expensive, procedure can be used to decompose a plethysm $\{\phi\} \circledcirc \{\psi\}$, for $\{\phi\}$ and $\{\psi\}$ characters (of tensor representations) of full linear groups, and corresponding to partitions of m_1 and m_2 respectively (with ψ a representation of $GL(k, F)$). Examining the total degrees (m_1 and m_2) of the Schur functions e_ϕ and e_ψ, and noting the effect of a plethysm as an operator on Schur functions, we find that the irreducible components of $\phi \circledcirc \psi$ are all irreducible characters corresponding to partitions of $m_1 m_2$. We can decompose the plethysm by evaluating it at at least as many matrices as there are irreducible (tensor) characters corresponding to these partitions; however, the cost of this method is prohibitive for all but very small values of k. Further algorithms for the decomposition of plethysms are discussed by Chen, Garsia, and Remmel [21].

Finally, we note that physical applications often involve characters of "restricted" subgroups of $GL(n, F)$ — such as the orthogonal group, the unitary group, or the symplectic group. The irreducible characters of these groups are closely related; *branching rules,* expressing an irreducible character of one of these groups in terms of irreducible characters of another, can be used for the evaluation and decomposition of characters of these restricted groups. Hamermesh [61] and Wybourne [119] each discuss these restrictive groups, and their representations and characters. Again, we can consider the problems of evaluating and decomposing characters of these groups. We leave the analysis of existing algorithms for these computations, and the search for new algorithms, for further work.

Final Remarks.

We have seen that a number of interesting computational problems arise from the structure theory of associative algebras and the representation theory of both finitely generated (and finite) groups, and the linear groups. While some ("representational") difficulties must be dealt with in order to work with the structures arising most often in physical applications (namely, algebras and matrix representations over $\mathbb{R}$ and $\mathbb{C}$), we have noted that in many cases the difficulties can be overcome. While symbolic (exact) computations appear to require more overhead than numerical computations, we have seen that they can be used to produce asymptotically fast (and, we hope, practical) algorithms for a number of problems which cannot be solved using strictly numerical techniques. In some cases, we have provided new algorithms for these problems; in others, it has been sufficient to provide analysis for the performance of algorithms which have been observed to work well in practice.

As we saw in Section 3, some of these representational problems disappear when we consider matrix representations and characters of finite groups — or decompose characters of linear groups obtained by forming products or plethysms of irreducible characters. It may be possible to provide correct and efficient numerical solutions for problems in these restricted domains, even though exact computations are necessary for these problems in a more general setting. For example, the problem of deciding whether a matrix representation over $\mathbb{R}$ of an arbitrary finitely generated group is isotypic cannot be solved reliably using a numerical estimate (of any fixed precision) for the input. However, if we consider a matrix representation of a (fixed) finite group G, then a reasonably accurate numerical estimate can be used (with a character table for G) to solve this problem correctly: The character of any representation of G is an (integer) linear combination of irreducible characters (see Dixon [35]). A study of those problems which can be solved numerically should prove to be rewarding — and there is certainly a wealth of numerical algorithms in the literature requiring analysis.

Since our symbolic algorithms are apparently more expensive than corresponding numerical algorithms, a "hybrid" approach may be worth considering, if symbolic representations of outputs are not required. In particular, it may be useful to begin by attempting to solve a problem numerically, simultaneously applying techniques from interval analysis to obtain a bound on the error in the numerical estimates computed. If the analysis indicates that the error may be too large, then symbolic techniques can be applied to obtain more accurate output. Otherwise, it is not necessary to continue. Again, we do not know whether numerical techniques for these problems can guarantee sufficiently accurate estimates (inexpensively), or whether interval analysis can produce sufficiently accurate estimates of error, for this approach to be more efficient than an immediate application of symbolic techniques. (For discussions of interval analysis, see [90] and [74].)

We have mentioned a number of problems, in Sections 2.6 and 3.4 which merit further attention. An examination of the references we list for representation theory (in particular, Curtis and Reiner [31], Hamermesh [61], and James and Kerber [69]) will yield numerous others, providing ample material for further work.

References

[1] L. M. Adleman and M. A. Huang.
Recognizing primes in random polynomial-time.
Proceedings, 19th ACM Symposium on Theory of Computing.
New York, 1987, 462–469.

[2] L. M. Adleman, C. Pomerance, and R. S. Rumley.
On distinguishing primes from composite numbers.
Annals of Mathematics 17 (1983), 173–206.

[3] A. V. Aho, J. E. Hopcroft, and J. D. Ullman.
The Design and Analysis of Computer Algorithms.
Addison-Wesley, Reading, 1974.

[4] D. S. Arnon, G. E. Collins, and S. McCallum.
Cylindrical algebraic decomposition I: the basic algorithm.
SIAM J. Comput. 13 (1984), 865–877.

[5] L. Babai and L. Rónyai.
Computing irreducible representations of finite groups.
Preprint, 1989. Extended abstract to appear in Proceedings,
30th Annual Symp. Foundations of Computer Science, 1989.

[6] E. Bach.
Fast algorithms under the extended Riemann Hypothesis:
a concrete estimate.
Proceedings, 14th Annual ACM Symposium on Theory of Computing,
San Francisco, 1982, 290–295.

[7] P. Beame, S. Cook, and H. J. Hoover.
Log depth circuits for division and related problems.
SIAM J. Comput. 15 (1986), 994–1003.

[8] R. E. Beck and B. Kolman (Editors).
Computers in Nonassociative Rings and Algebras.
Academic Press, New York, 1977.

[9] S. J. Berkowitz. On computing the determinant in small parallel time using a small number of processors.
Information Processing Letters 18 (1984), 147–150.

[10] E. R. Berlekamp. Factoring polynomials over finite fields.
Bell System Tech. J. 46 (1967), 1853–1859.

[11] E. R. Berlekamp. Factoring polynomials over large finite fields.
Math. Comp. 24 (1970), 713–735.

[12] E. Bishop and D. Bridges. *Constructive Analysis.*
Springer-Verlag, Berlin, New York, 1985.

[13] A. Borodin, S. Cook, and N. Pippenger.
Parallel computation for well-endowed rings and space-bounded probabilistic machines.
Information and Control 58 (1983), 113–136.

[14] A. Borodin, J. von zur Gathen, and J. Hopcroft.
Fast parallel matrix and GCD computations.
Information and Control 52 (1982), 241–256.

[15] B. Buchberger, G. E. Collins, and R. Loos., ed.
Computer Algebra. Symbolic and Algebraic Computation.
Springer, New York, 1983 (2nd Edition).

[16] G. Butler. Computational Group Theory.
Technical Report 276, Department of Computer Science, University of Sydney, 1986.

[17] J. J. Cannon. Computers in group theory: A survey.
Communications of the ACM 12 (1969), 3–12.

[18] D. G. Cantor, and E. Kaltofen.
Fast multiplication of polynomials over arbitrary rings.
Preliminary report, 12 pp., 1987.

[19] D. G. Cantor, and H. Zassenhaus.
A new algorithm for factoring polynomials over finite fields.
Math. Comp. 36 (1981), 587–592.

[20] Chen Jingrun.
On the least prime in an arithmetical progression and theorems concerning the zeros of Dirichlet's L-functions (II).
Scientia Sinica 22 (1979), 859–889.

[21] Y. M. Chen, A. M. Garsia, and J. Remmel. Algorithms for plethysm.
In *Contemporary Mathematics No. 34: Combinatorics and Algebra.*
American Mathematical Society, 1984, 109–153.

[22] A. L. Chistov. Fast parallel calculation of the rank of matrices over a field of arbitrary characteristic.
Proc. Int. Conf. Foundations of Computation Theory,
Lecture Notes in Computer Science 199, Springer, New York, 1985, 63–69.

[23] S. Chowla. On the least prime in an arithmetical progression.
Journal, Indian Mathematical Society 1 (1934), 1–3.

[24] G. E. Collins. Quantifier elimination in real closed fields by cylindrical algebraic decomposition. In *Automata Theory and Formal Languages.* 2nd GI Conference, Kaiserslautern, May 1975. Lecture Notes in Computer Science 33, Springer-Verlag, Berlin, 1975, 134–183.

[25] G. E. Collins. Infallible calculation of polynomial zeros to specified precision. In *Mathematical Software* III. (J. R. Rice, ed.) Academic Press, New York, 1977, 35–68.

[26] G. E. Collins and R. Loos. Real zeros of polynomials. In *Computer Algebra. Symbolic and Algebraic Computation*([15]), 1983, 83–94.

[27] G. E. Collins, M. Mignotte, and F. Winkler. Arithmetic in basic algebraic domains. In *Computer Algebra. Symbolic and Algebraic Computation*([15]), 1983, 189–220.

[28] S. A. Cook. A taxonomy of problems with fast parallel algorithms. Information and Control 64 (1985), 2–22.

[29] D. Coppersmith and S. Winograd. Matrix multiplication via arithmetic progressions. Proceedings, 19th Annual ACM Symposium on Theory of Computing, New York, 1987, 1–6.

[30] L. Csanky. Fast parallel matrix inversion algorithms. SIAM J. Comp. 5 (1976), 618–623.

[31] C. W. Curtis and I. Reiner. *Representation Theory of Finite Groups and Associative Algebras.* Wiley, New York, 1962.

[32] H. Davenport. *Multiplicative Number Theory.* Second Edition. Springer-Verlag, New York, Berlin, 1980.

[33] L. E. Dickson. *Algebras and Their Arithmetics.* The University of Chicago Press, Chicago, 1923.

[34] J. D. Dixon. High speed computation of group characters. Numerische Mathematik 10 (1967), 446–450.

[35] J. D. Dixon. Computing irreducible representations of groups. Math. Comp. 24 (1970), 707–712.

[36] J. D. Dixon. Asymptotically fast factorization of integers. Math. Comp. 36 (1981), 255–260.

[37] W. Eberly. Very fast parallel matrix and polynomial arithmetic.
Technical Report 178/85, Department of Computer Science,
University of Toronto, 1985. Extended Abstract in Proceedings,
25th Annual Symp. Foundations of Computer Science (1984), 21–30.

[38] W. Eberly. Very fast parallel polynomial arithmetic.
Preprint, 1989. To appear in SIAM J. Comput.

[39] O. Eğecioğlu. Algorithms for the character theory of the symmetric group.
Proceedings, Eurocal '85, Volume 2. Lecture Notes in Computer Science,
Springer-Verlag, 1985, 206–224.

[40] V. Felsch.
A bibliography on the use of computers in group theory and related topics:
algorithms, implementations, and applications.
SIGSAM Bulletin 12 (1978), 23–86.

[41] F. Fich and M. Tompa.
The parallel complexity of exponentiating polynomials over finite fields.
JACM 35 (1988), 651–667.

[42] H. O. Foulkes. Plethysm of S-functions.
Royal Soc. London, Phil Trans. A 246 (1954), 555–591.

[43] K. Friedl and L. Rónyai.
Polynomial time solutions for some problems in computational algebra.
Proceedings, 17th ACM Symposium on Theory of Computing, Providence,
1985, 153–162.

[44] A. Fröhlich and J. C. Shepherdson. Effective procedures in field theory.
Royal Soc. London, Phil. Trans. A 248 (1955-56), 407–432.

[45] J. Gabriel.
New methods for reduction of group representations using an extension
of Schur's lemma.
J. Math. Phys. 5 (1964), 494–504.

[46] J. Gabriel. New methods for reduction of group representations. II
J. Math. Phys. 9 (1968), 973–976.

[47] J. Gabriel. New methods for reduction of group representations. III
J. Math. Phys. 10 (1969), 1789–1795.

[48] J. Gabriel. New methods for reduction of group representations. IV
J. Math. Phys 10 (1969), 1932–1934.

[49] J. Gabriel. Numerical methods for reduction of group representations.
Proceedings, 2nd ACM Symp. Symbolic and Algebraic Manipulation,
Los Angeles, 1971, 180–182.

[50] Z. Galil and V. Pan. Improved processor bounds for algebraic and combinatorial problems in RNC.
Proceedings, 26th Annual Symp. Foundations of Computer Science,
Portland, 1985, 490–495.

[51] M. R. Garey and D. S. Johnson.
Computers and Intractibility. A Guide to the Theory of NP-Completeness.
W. H. Freeman and Company, San Francisco, 1979.

[52] J. von zur Gathen. Parallel algorithms for algebraic problems.
SIAM J. Comput. 13 (1984), 802–824.

[53] J. von zur Gathen.
Representations and parallel computations for rational functions.
SIAM J. Comput. 15 (1986), 432–452.

[54] J. von zur Gathen. Parallel arithmetic computations.
Proceedings, 12th Int. Symp. Math. Foundations of Computer Science,
Bratislaava, Springer Lecture Notes in Computer Science **233**, 1986, 93–112.

[55] J. von zur Gathen. Computing powers in parallel.
SIAM J. Comput. 16 (1987), 930–945.

[56] J. von zur Gathen. Algebraic complexity theory.
Annual Review of Computer Science 3 (1988), 317–347.

[57] J. von zur Gathen. Inversion in finite fields using logarithmic depth.
J. Symb. Comp., to appear.

[58] J. von zur Gathen and G. Seroussi.
Boolean circuits versus arithmetic circuits.
Proceedings, 6th Int. Conf. Computer Science, Santiago, Chile, 1986,
171–184.

[59] S. Goldwasser and S. Micali. Probabilistic encryption.
JCSS 28 (1984), 270-299.

[60] S. Graham. On Linnik's constant.
Acta Arithmetica 39 (1981), 163–179.

[61] M. Hamermesh. *Group Theory and Its Application to Physical Problems.*
Addison-Wesley, Reading, 1962.

[62] I. N. Herstein. *Noncommutative Rings.*
Math. Association of America, 1968.

[63] H. J. Hoover. Feasibly constructive analysis.
Technical Report 206/87, Department of Computer Science,
University of Toronto, 1987.

[64] Hua Loo Keng. *Introduction to Number Theory.*
Springer-Verlag, Berlin, 1982.

[65] O. H. Ibarra, S. Moran, and L. E. Rosier.
A note on the parallel complexity of computing the rank of order n matrices.
Information Processing Letters 11 (1980), 162.

[66] N. Jacobson. *Lie Algebras.*
Wiley, New York, 1962.

[67] N. Jacobson. *Basic Algebra I.*
W. H. Freeman, San Francisco, 1974.

[68] N. Jacobson. *Basic Algebra II.*
W. H. Freeman, San Francisco, 1980.

[69] G. James and A. Kerber.
The Representation Theory of the Symmetric Group.
Encyclopedia of Mathematics and Its Applications, Volume 16.
Addison-Wesley, Reading, 1981.

[70] E. Kaltofen.
Computing with polynomials given by straight-line programs I:
greatest common divisors.
Proceedings, 17th Annual Symp. Theory of Computing,
Providence, 1985, 131–142.

[71] E. Kaltofen. Uniform closure properties of p-computable functions.
Proceedings, 18th Annual Symp. Theory of Computing,
Berkeley, 1986, 330–337.

[72] E. Kaltofen, M. Krishnamoorthy, and B. D. Saunders.
Fast parallel algorithms for similarity of matrices.
Proceedings, 1986 ACM Symp. Symbolic and Algebraic Computations, 1986,
65–70.

[73] S. L. Kleinman.
Problem 15. Rigorous foundation of Schubert's enumerative calculus.
In *Mathematical Developments Arising from the Hilbert Problems*
Proceedings of Symposia in Pure Mathematics, Volume 28 (1976), 445–482.

[74] D. E. Knuth.
The Art of Computer Programming. Vol. 2: Seminumerical Algorithms.
Second Edition. Addison-Wesley, Reading, 1981.

[75] K. Ko and H. Friedman. Computational complexity of real functions.
Theoret. Comput. Sci. 20 (1982), 323–352.

[76] S. Landau. Factoring polynomials over algebraic number fields.
SIAM J. Comput. 14 (1985), 184–195.

[77] S. Landau. Factoring polynomials quickly.
Not. Am. Math. Soc. 34 (1987), 3–8.

[78] A. Lascoux and M.-P. Schützenberger.
Schubert polynomials and the Littlewood-Richardson rule.
Letters in Mathematical Physics 10 (1985), 111–124.

[79] J. W. Leech and D. J. Newman. *How to Use Groups.*
Methuen, London, 1969.

[80] A. Lempel, G. Seroussi, and J. Ziv.
On the power of straight-line computations in finite fields.
IEEE Transactions on Information Theory 6 (1982), 875–880.

[81] A. K. Lenstra. Factoring polynomials over algebraic number fields.
In *Computer Algebra.* Proceedings, EUROCAL, London, England,
March 1983. (J. A. van Hulzen, ed.) Springer, New York, 1983, 245–254.

[82] A. K. Lenstra, H. W. Lenstra, and L. Lovász.
Factoring polynomials with rational coefficients.
Math. Ann. 261 (1982), 515–534.

[83] U. V. Linnik.
On the least prime in an arithmetic progression.
I. The basic theorem.
Matematicheskiĭ Sbornik 57 (1944), 139–178.

[84] U. V. Linnik.
On the least prime in an arithmetic progression.
II. The Deuring-Heilbronn phenomenon.
Matematicheskiĭ Sbornik 57 (1944), 347–367.

[85] B. E. Litow and G. I. Davida. $O(\log(n))$ parallel time finite field inversion.
In Proc. 3$^{\text{rd}}$ Aegean Workshop on Computing, Corfu, Lecture Notes in Computer Science 319, Springer-Verlag, Berlin, New York, 1988, pp. 74–80.

[86] D. E. Littlewood and A. R. Richardson. Group characters and algebra.
Royal Soc. London, Phil. Trans. A 233 (1934), 99–141.

[87] R. Loos. Computing in algebraic extensions.
In *Computer Algebra. Symbolic and Algebraic Computation*([15]), 1983,
173–187.

[88] I. G. Macdonald. *Symmetric Functions and Hall Polynomials.*
Oxford University Press, Oxford, 1979.

[89] M. Mignotte. Some useful bounds.
In *Computer Algebra. Symbolic and Algebraic Computation*([15]), 1983, 259–263.

[90] R. E. Moore. *Methods and Applications of Interval Analysis.*
SIAM, Philadelphia, 1979.

[91] K. Mulmuley. A fast parallel algorithm to compute the rank of a matrix over an arbitrary field.
Combinatorica 7 (1987), 101–104.

[92] F. D. Murnaghan. *The Theory of Group Representations.*
The John Hopkins Press, 1938; Dover, New York, 1963.

[93] J. Neubüser. Computing with groups and their character tables.
In *Computer Algebra. Symbolic and Algebraic Computations*([15]), 1983, 45–56.

[94] V. Pan and J. Reif.
Efficient parallel solution of linear systems.
Proceedings, 17th Annual ACM Symp. Theory of Computing,
Providence, 1985, 143–152.

[95] R. S. Pierce. *Associative Algebras.*
Springer-Verlag, New York, 1982.

[96] J. R. Pinkert. An exact method for finding the roots of a complex polynomial.
ACM Transactions on Mathematical Software 2 (1976), 351–363.

[97] C. Pomerance.
A note on the least prime in an arithmetic progression.
Journal of Number Theory 12 (1980), 218–223.

[98] M. Rabin. Digitalized signatures and public-key functions as intractible as factorization.
MIT/LCS TR 212, Technical Memo MIT, 1979.

[99] M. Rabin. Probabilistic algorithms in finite fields.
SIAM J. Comput. 9 (1980), 278–280.

[100] J. H. Reif. Logarithmic depth circuits for algebraic functions.
SIAM J. Comput. 15 (1986), 231–242.

[101] J. B. Remmel and R. Whitney. Multiplying Schur functions.
Journal of Algorithms 5 (1984), 471–487.

[102] L. Rónyai. Zero divisors and invariant subspaces.
Technical Report CIS-TR 85-11, Department of Computer Science,
University of Oregon, 1985.

[103] L. Rónyai. Simple algebras are difficult.
Proceedings, 19th ACM Symposium on Theory of Computing,
New York, 1987, 398–408.

[104] L. Rónyai. Zero divisors in quaternion algebras.
Journal of Algorithms 9 (1988), 494–506.

[105] J. B. Rosser and L. Schoenfeld.
Approximate functions for some functions of prime numbers.
Illonois J. Math. 6 (1962), 64–94.

[106] W. L. Ruzzo. On uniform circuit complexity.
J. Computer System Sciences 22 (1981), 365–383.

[107] J. E. Savage. *The Complexity of Computing.*
Wiley, New York, 1976.

[108] G. J. A. Schneider. Dixon's character table algorithm revisited.
J. Symb. Comp., 1989, to appear.

[109] A. Schönhage.
Schnelle Multiplikation von Polynomen über Körpen der Charakteristik 2.
Acta. Inf. 7 (1977), 395–398.

[110] A. Schönhage.
The fundamental theorem of algebra in terms of computational complexity.
Technical Report, Universität Tübingen, 1982.

[111] J. T. Schwartz.
Fast probabilistic algorithms for verification of polynomial identities.
J. ACM 27 (1980), 701–717.

[112] J.-P. Serre. *Linear Representations of Finite Groups.*
Springer-Verlag, New York, 1977.

[113] R. P. Stanley. Theory and application of plane partitions: part 1.
Studies in Applied Mathematics 50 (1971), 167–188.

[114] V. Strassen. Gaussian elimination is not optimal.
Numer. Mathematik 13 (1969), 354–356.

[115] E. C. Titchmarsch. A divisor problem.
Rendiconti del Circolo Matematico di Palermo 54 (1930), 414–429.

[116] J. A. Todd. A note on the algebra of S-functions.
Cambridge Philosophical Society, Proceedings, 45 (1949), 328–334.

[117] B. L. van der Waerden. *Algebra,* Volume 1.
Ungar, New York, 1970.

[118] B. L. van der Waerden. *Algebra,* Volume 2.
Ungar, New York, 1970.

[119] B. G. Wybourne. *Symmetry Principles and Atomic Spectroscopy.*
Wiley, New York, 1970.