Asymptotically Efficient Algorithms for the Frobenius Form

Wayne Eberly∗
Department of Computer Science
University of Calgary
Calgary, Alberta, Canada T2N 1N4
Email: eberly@cpsc.ucalgary.ca
http://www.cpsc.ucalgary.ca/~eberly

May 1, 2000

Abstract

A new randomized algorithm is presented for computation of the Frobenius form of an $n \times n$ matrix over a field. A version of the algorithm is presented that uses standard arithmetic whose asymptotic expected complexity matches the worst case complexity of the best known deterministic algorithm for this problem, recently given by Storjohann and Villard [25], and that seems to be superior when applied to sparse or structured matrices with a small number of invariant factors. A version that uses asymptotically fast matrix multiplication is also presented. This is the first known algorithm for this computation over small fields whose asymptotic complexity matches that of the best algorithm for computations over large fields and that also provides a Frobenius transition matrix over the ground field.

As an application, it is shown that a “rational Jordan form” of an $n \times n$ matrix over a finite field can also be computed asymptotically efficiently.

1 Introduction

The computation of a normal form for an $n \times n$ matrix $A$ over a field $F$ is a classical mathematical problem. It is well known (see, for example, Gantmacher [9]) that every matrix $A \in F^{n \times n}$ is similar to a unique block diagonal matrix with companion matrices of monic polynomials $f_1, f_2, \ldots, f_k$ on the diagonal, where $f_i$ is divisible by $f_{i+1}$ for $1 \leq i \leq k - 1$. That is, there exists a nonsingular matrix $V \in F^{n \times n}$ such that

$$ VAV^{-1} = F_A = \begin{bmatrix} C_{f_1} & 0 & \cdots & 0 \\ 0 & C_{f_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & C_{f_k} \end{bmatrix} $$

(1)

and where

$$ C_g = \begin{bmatrix} 0 & \cdots & 0 & -g_0 \\ 1 & 0 & -g_1 \\ \cdots & \cdots & \cdots \\ 0 & 1 & -g_{d-1} \end{bmatrix} \in F^{d \times d} $$

∗Research was supported in part by the Natural Sciences and Engineering Research Council of Canada.
is the companion matrix of a monic polynomial \( g = x^d + g_{d-1}x^{d-1} + g_{d-2}x^{d-2} + \cdots + g_1x + g_0 \in F[x] \)
with degree \( d \). The above matrix \( F_A \) is now commonly called the Frobenius form of \( A \), and the
polynomials \( f_1, f_2, \ldots, f_k \) are called the invariant factors of \( A \). The first invariant factor, \( f_1 \), is also
the minimum polynomial of \( A \), and the characteristic polynomial of \( A \) is the product \( \prod_{i=1}^{k} f_i \).

If we insist (as usual) that the degree of \( f_k \) be positive then the invariant factors are unique;
we will call \( k \) the number of (nontrivial) invariant factors of the matrix \( A \). The above matrix \( V \) is
not unique; every nonsingular matrix \( V \) satisfying equation (1), above, will be called a Frobenius transition matrix
for \( A \).

Several deterministic algorithms for computation of the Frobenius form and a transition matrix
are known; see, in particular, Ozello [22], Lüneburg [19], and, more recently, Augot and Camion [1]
and Steel [23]. Augot and Camion also provide evidence that the number of invariant factors is
typically small.

A randomized algorithm that is asymptotically much more efficient than any of these has been
given by Giesbrecht [11], [12]: Giesbrecht’s Las Vegas algorithm can be used to compute both
the Frobenius form and a Frobenius transition matrix for a given matrix \( A \in F^{n \times n} \) over a field \( F \)
using an expected number of operations over \( F \) that is in \( O(n^3) \), with standard matrix and polyno-
mial arithmetic, whenever \( F \) has at least \( n^2 \) distinct elements, and using an expected number of
operations in \( O(n^3 \log_q n) \) if \( F \) is a finite field with size \( q \).

If asymptotically fast matrix and polynomial arithmetic are used then these results can be im-
proved. Suppose, in particular, that it is possible to compute the product of two \( n \times n \) matrices
using \( O(MM(n)) \) operations over \( F \). One can take \( MM(n) \) to be \( n^{\log_2 7} \leq n^{2.81} \) using the algo-
rithm of Strassen [26], while the algorithm of Coppersmith and Winograd [8] gives the best known
asymptotic result, with \( MM(n) \leq n^{4.76} \). Giesbrecht’s asymptotically fast algorithm can be used
to compute the Frobenius form and a Frobenius transition matrix of a given matrix \( A \in F^{n \times n} \) using
an expected number of operations in \( O(MM(n) \log n) \) whenever \( F \) has at least \( n^2 \) elements.

One can compute the Frobenius form of a matrix over a smaller field that is within a polylog
factor of this bound, by performing computations over a small field extension, because the normal
form is unique and guaranteed to be a matrix over a ground field. This trick cannot generally be
used to find a transition matrix in the ground field as well. Thus, Giesbrecht’s work leaves open
the question of whether one could find a Frobenius transition matrix asymptotically quickly.

More recently, Storjohann [24] has given a deterministic algorithm to compute the Frobenius
form of a matrix \( A \in F^{n \times n} \) over an arbitrary field \( F \) using \( O(n^3) \) operations in the worst case,
with standard matrix and polynomial arithmetic. Storjohann and Villard [25] have extended this
algorithm to compute a Frobenius transition matrix at this cost as well — matching the expected
cost of Giesbrecht’s randomized algorithm, under standard arithmetic, for computations over large
fields, and improving the complexity by a log factor for computations over small fields.

In this paper, another new randomized algorithm is presented for this computation. A version
of the algorithm that uses standard arithmetic has an expected cost that is asymptotically the same
as the worst case cost recently achieved by Storjohann [24] and Storjohann and Villard [25]. The new
algorithm slightly extends and adapts techniques that were used by Wiedemann [28] to compute
the minimum polynomial, and that were used by Lambert [17] to produce a version of Lanczos’
algorithm for computations over finite fields. Like Wiedemann’s algorithm, the new (standard
arithmetic) algorithm is to some extent a “black box” algorithm: It requires an expected number
of \( O(n) \) multiplications of the given matrix \( A \) by vectors, \( O(n) \) multiplications of the transpose
\( A^T \) by vectors, and \( O(kn^2) \) operations over \( F \), where \( k \) is the number of invariant factors of the
matrix. It also requires \( O(n^2) \) storage space. Thus, while it fails to match either the time-
or space-bounds that are sometimes associated with “black box” algorithms for matrix computations,
its expected complexity is subcubic if both the input matrix $A$ is sparse or structured, so that the cost of multiplication of $A$ and $A^T$ by a vector is subquadratic in $n$, and the number $k$ of invariant factors of $A$ is sublinear in $n$. The new algorithm also generates data that allows one to solve a given system $Vx = y$, for the generated Frobenius transition matrix $V$, quite efficiently, as one might wish to when the Frobenius form is being applied.

A version of the new algorithm that uses asymptotically fast matrix multiplication computes both the Frobenius form and a Frobenius transition matrix, using an expected number of operations over any field $F$ that is in $O(\mathcal{M}\mathcal{M}(n) \log n)$, under the common assumption that $\mathcal{M}\mathcal{M}(n) \in \Omega(n^{2+\epsilon})$ for some positive real number $\epsilon$. This reduces the cost needed to compute the Frobenius form over a small field by the extra log factors needed to implement Giesbrecht’s algorithm over a field extension and, to my knowledge, demonstrates for the first time that a Frobenius transition matrix over the ground field can be computed in subcubic time, in the small field case.

Quite recently, Villard [27] has extended the Krylov-based techniques used by Lanczos, Wiedemann and others in a different way, through the application of low rank conditioners, to obtain a new black box algorithm for the invariant factors of a matrix: If a matrix $A \in F^{n \times n}$ has at most $\mu$ distinct invariant factors and the field $F$ is sufficiently large, then the invariant factors of $A$ can be computed by a Monte Carlo algorithm, using $O(\mu n \log n)$ multiplications of $A$ by vectors and using $O(\mu^2 n \log \log n)$ additional operations over $F$. The computation can be performed over a small finite field by working over a field extension, increasing the number of multiplications of $A$ by vectors (over the ground field) by a factor of $O(\log n)$, and increasing the number of additional operations over $F$ by a small polylog factor. Since $\mu \in O(\sqrt{n})$ and, as noted above, the characteristic polynomial of $A$ is the product of the invariant factors, this provides the first subcubic black box algorithm, using standard arithmetic, for the characteristic polynomial of sparse or structured matrices, and constitutes significant progress toward the development of a black box algorithm for the characteristic polynomial — see Kaltofen [13, Open Problem 3] and Villard [27] for a discussion of this topic and additional references.

As noted above, the techniques used in this paper are adaptations of methods that have been used to compute minimum polynomials (and cyclic vectors) for matrices. The relevant definitions and results concerning minimum polynomials are presented here in Section 2. These methods are extended, so that they can be used to compute all invariant factors and corresponding columns of a Frobenius transition matrix, in Section 3. The new algorithm is presented and analyzed in Section 4. Finally, it is applied to compute a “rational Jordan form” of a matrix over a finite field, asymptotically efficiently, in Section 5.

2 Minimum Polynomials

In this section, results from the literature concerning the computation of minimum polynomials of linear recurrences, matrices and vectors, and matrices are generalized to apply when matrices are considered as linear operators on subspaces.

Subsection 2.1 introduces minimum polynomials of sequences, as well as some associated values that will be of interest, and reviews results from the literature about the complexity of computing them. A few straightforward improvements, that one can make when a bound on the degree of the desired minimum polynomial is available, are also noted. Subsection 2.2 introduces minimum polynomials of matrices and vectors, and includes minor generalizations of results of Wiedemann [28] and Kaltofen and Pan [14] concerning the complexity of computing them. This is continued in Subsection 2.3, which introduces minimum polynomials of matrices and subspaces.
2.1 Minimum Polynomials of Sequences

Let $F$ be a field, and suppose $A \in F^{n \times n}$ and that $u$ and $v$ are vectors in $F^{n \times 1}$, so that $u^T A^i v \in F$ for every integer $i \geq 0$.

**Definition 2.1.** The minimum polynomial of the linear recurrence $u^T v, u^T A v, u^T A^2 v, u^T A^3 v, \ldots$, denoted $\text{minpol}(u^T, A, v)$, is the monic polynomial $x^m + c_{m-1} x^{m-1} + \cdots + c_1 x + c_0$ of least degree such that

$$u^T A^{m+i} v + c_{m-1} u^T A^{m-1+i} v + \cdots + c_1 u^T A^i v + c_0 u^T A^i v = 0$$

for every integer $i \geq 0$.

Note that, by linearity, $\text{minpol}(u^T, A, v)$ is also the monic polynomial $f$ of least degree such that $u^T s(A) f(A) v = 0$ for every polynomial $s \in F[x]$.

**Fact 2.2.** Let $A$, $u$, and $v$ be as above.

(a) Given $A$, $u$, and $v$, it is possible to compute $\text{minpol}(u^T, A, v)$ deterministically, by computing the product of $A$ and each of $O(n)$ vectors, and performing $O(n^2)$ additional operations over $F$.

(b) Given $A$, $u$, $v$, and an integer $k \leq n$ such that the degree of $\text{minpol}(u^T, A, v)$ is less than or equal to $k$, it is possible to compute $\text{minpol}(u^T, A, v)$ deterministically, by computing the product of $A$ and each of $O(k)$ vectors, and performing $O(k n)$ additional operations over $F$.

**Proof.** If $A$, $u$, and $v$ are known then $\text{minpol}(u^T, A, v)$ can be computed as described in part (a) in several ways, including by an application of the Berlekamp-Massey algorithm [2, 20, 21] or a transpose-free version of the Lanczos process [7].

If a bound $k$ on the degree of the minimum polynomial is also known, then one can take advantage of the fact that $\text{minpol}(u^T, A, v)$ is determined from the first $2k$ entries of the linear recurrence,

$$u^T v, u^T A v, u^T A^2 v, \ldots, u^T A^{2k-1} v$$

— see Lemma 1 of Kaltofen and Pan [14] for additional details and a proof. This implies that the minimum polynomial is available, when either of the above methods has been applied, after these entries of the recurrence have been processed. The complexity bounds stated above in part (b) follow by a straightforward analysis of these algorithms.  

Algorithms presented in the sequel will make use of a function $\text{minpolseq}(u^T, A, v, k)$ that receives the transpose of a vector $u \in F^{n \times 1}$, a matrix $A \in F^{n \times n}$, vector $v \in F^{n \times 1}$, and a positive integer $k$ such that $\text{minpol}(u^T, A, v)$ has degree at most $k$ as input, and that returns $\text{minpol}(u^T, A, v)$ as output. It will be assumed that the cost of executing this function is bounded as described in Fact 2.2(b), above.

Suppose now that $\text{minpol}(u^T, A, v)$ has degree $m$ for some positive integer $m \leq n$. Let $K_{u,v}^{(L)}$ denote the vector space spanned by the vectors

$$u, (A^T)u, (A^T)^2 u, \ldots, (A^T)^{m-1} u$$

and let $K_{u,v}^{(R)}$ denote the vector space spanned by the vectors

$$v, Av, A^2 v, \ldots, A^{m-1} v.$$
Definition 2.3. Two sequences of vectors \( u_1, u_2, \ldots, u_m \in K_{u,v}^{(L)} \) and \( v_1, v_2, \ldots, v_m \in K_{u,v}^{(R)} \) form dual bases for \( A, u \) and \( v \) if
\[
 u_i^T v_j \neq 0 \quad \text{if and only if} \quad i = j
\]
for \( 1 \leq i, j \leq m \).

Note that if such sequences exist at all, then equation (2) can be used to establish that \( u_1, u_2, \ldots, u_m \) are linearly independent in \( K_{u,v}^{(L)} \) and that \( v_1, v_2, \ldots, v_m \) are linearly independent in \( K_{u,v}^{(R)} \). A comparison of the number of vectors in each sequence with the dimension of the each vector space confirms that \( u_1, u_2, \ldots, u_m \) forms a basis for \( K_{u,v}^{(L)} \) and that \( v_1, v_2, \ldots, v_m \) forms a basis for \( K_{u,v}^{(R)} \), as the name “dual bases” suggests. Therefore, the next definition generalizes the last one.

Definition 2.4. If \( U, V \) are subspaces of \( F^{n \times 1} \) that each have dimension \( m \) over \( F \) then two sequences of vectors \( u_1, u_2, \ldots, u_m \) and \( v_1, v_2, \ldots, v_m \) form dual bases for \( A, U \), and \( V \) if \( u_1, u_2, \ldots, u_m \) is a basis for \( U \) over \( F \), \( v_1, v_2, \ldots, v_m \) is a basis for \( V \) over \( F \), and
\[
 u_i^T v_j \neq 0 \quad \text{if and only if} \quad i = j
\]
for \( 1 \leq i, j \leq m \).

Fact 2.5. Let \( A \in F^{n \times n}, u, \) and let \( u, v \in F^{n \times 1} \).

(a) Given the matrix \( A \) and vectors \( u \) and \( v \), it is possible to produce dual bases for \( A, u \) and \( v \) deterministically, by computing the product of \( A \) and \( O(n) \) vectors, the product of \( A^T \) and \( O(n) \) vectors, and performing \( O(n^2) \) additional operations over \( F \).

(b) Given \( A, u, v, \) and an integer \( k \leq n \) such that \( k \) is greater than or equal to the degree \( m \) of \( \text{minpol}(u^T, A, b) \), it is possible to produce dual bases for \( A, u \) and \( v \) deterministically, by computing the product of \( A \) and \( O(k) \) vectors, the product of \( A^T \) and \( O(k) \) vectors, and performing \( O(kn) \) additional operations over \( F \).

Proof. A “bi-orthogonal Lanczos algorithm with lookahead,” as described, for example, by Lambert [17], can be used to perform the computation described in part (a). Once again, if a bound \( k \) on the degree of the minimum polynomial of \( \text{minpol}(u^T, A, v) \) is available, then one can safely terminate this process as soon as it is realized that further computations would either lead to an “incurable breakdown” or a sequence of more than \( k \) linearly independent vectors in \( K_{u,v}^{(L)} \). This can be checked by keeping track of the number of elements of dual bases generated so far, as well as the length of any lookahead stage currently in progress, so that the entire computation has the cost described in part (b) above. \( \Box \)

Algorithms presented in the sequel will make use a function \( \text{dualbasis}(u^T, A, v, k) \) that receives the transpose of a vector \( u \in F^{n \times 1} \), a matrix \( A \in F^{n \times n} \), vector \( v \in F^{n \times 1} \), and a positive integer \( k \) such that \( \text{minpol}(u^T, A, v) \) has degree at most \( k \) as input, and that returns dual bases for \( A, u \) and \( v \) as output. It will be assumed that the cost of executing this function is bounded as described in Fact 2.5(b).
2.2 Minimum Polynomials of Matrices and Vectors

**Definition 2.6.** A subspace $U$ of the vector space $F^{n \times 1}$ is $A$-invariant (for a given matrix $A \in F^{n \times n}$) if $Au \in U$ for all $u \in U$.

**Definition 2.7.** Let $U \subseteq F^{n \times 1}$ be $A^T$-invariant for a matrix $A \in F^{n \times n}$, and let $v \in F^{n \times 1}$. Then $U^T \subseteq F^{1 \times n}$ is the set of transposes $u^T$ of vectors $u$ in $U$, and the minimum polynomial of $U^T$, $A$, and $v$, denoted $\text{minpol}(U^T, A, v)$, is the monic polynomial $x^m + c_{m-1}x^{m-1} + \cdots + c_1x + c_0$ of least degree such that

$$u^T A^m v + c_{m-1}u^T A^{m-1} v + \cdots + c_1u^T Av + c_0 u^T v = 0$$

for every element $u$ of $U$.

Note that if $U$ is $A^T$-invariant then $(A^T)^i u \in U$ for every element $u$ of $U$ and every integer $i \geq 0$. Since $((A^T)^i u)^T = u^T A^i$, it follows that $\text{minpol}(U^T, A, v)$ is also the monic polynomial $f$ of least degree such that $u^T A^i f(A)v = 0$ for every element $u$ of $U$ and every integer $i \geq 0$, and also (by linearity) the monic polynomial $f$ of least degree such that $u^T s(A)f(A)v = 0$ for every element $u$ of $U$ and every polynomial $s \in F[x]$.

One can define the minimum polynomial $\text{minpol}(A, v)$ of a sequence of vectors $v, Av, A^2v, \ldots$ to be the monic polynomial $x^m + c_{m-1}x^{m-1} + \cdots + c_1x + c_0$ of least degree such that

$$A^m v + c_{m-1}A^{m-1} v + \cdots + c_1 Av + c_0 = 0. \quad (3)$$

Multiplying both sides of this equation by $u^T A^i$ on the left, one has that

$$u^T A^{m+i} v + c_{m-1}u^T A^{m-1+i} v + \cdots + c_1u^T A^{i+1} v + c_0 u^T A^i v = 0 \quad \text{for all } u \in U, \quad (4)$$

and for every nonnegative integer $i$, so that condition (3) implies condition (4). On the other hand, if condition (4) is satisfied for every vector $u$ in a basis for $F^{n \times 1}$ then condition (3) is satisfied too, so that these conditions are equivalent and

$$\text{minpol}(A, v) = \text{minpol}(F^{1 \times n}, A, v).$$

It is also easily checked for any given matrix $A \in F^{n \times n}$, $A^T$-invariant subspace $U$ of $F^{n \times 1}$, and vector $v \in F^{n \times 1}$, that

$$\text{minpol}(U^T, A, v) = \text{lcm}_{u \in U}(\text{minpol}(u^T, A, v)).$$

Thus $\text{minpol}(U^T, A, v)$ is a divisor of $\text{minpol}(A, v)$, since $\text{minpol}(u^T, A, v)$ is, for every vector $u$.

Wiedemann [28] has presented an algorithm to compute the minimum polynomial $\text{minpol}(A, v)$ of a matrix and vector over a finite field $F$, as the least common multiple of minimum polynomials $\text{minpol}(u^T, A, v)$, for a constant number of uniformly and independently selected vectors $u \in F^{n \times 1}$. Results of Kaltofen and Pan [14] establish a similar result for computations over large fields as well. Their techniques, and Fact 2.2, can be used to establish the following.

**Fact 2.8.** Let $A \in F^{n \times n}$ and let $v \in F^{n \times 1}$. Then the following computations can be performed at the stated cost, using a Las Vegas algorithm that either returns the desired output (with probability at least one-half) or reports failure.

(a) If the field $F$ is finite then $\text{minpol}(A, v)$ can be computed by uniformly and independently selecting $O(n)$ elements from $F$, computing the product of $A$ and $O(n)$ vectors, and performing $O(n^2)$ additional operations over $F$. 

6
If the field \( F \) is infinite and \( S \) is a finite subset of \( F \) including at least \( 2n \) distinct elements, then \( \text{minpol}(A,v) \) can be computed by uniformly and independently selecting \( n \) elements from \( S \), computing the product of \( A \) and \( O(n) \) vectors, and performing \( O(n^2) \) additional operations over \( F \).

(b) If one is also given an integer \( k \leq n \) such that the degree of \( \text{minpol}(A,v) \) is less than or equal to \( k \), and the field \( F \) is finite, then \( \text{minpol}(A,v) \) can be computed by uniformly and independently selecting \( O(n) \) elements from \( F \), computing the product of \( A \) and \( O(k) \) vectors, and performing \( O(kn) \) additional operations over \( F \).

If the above integer \( k \) is also given and \( F \) is infinite, and \( S \) is a finite subset of \( F \) containing at least \( 2k \) distinct elements, then \( \text{minpol}(A,v) \) can be computed by uniformly and independently selecting \( n \) elements from \( S \), computing the product of \( A \) and \( O(k) \) vectors, and performing \( O(kn) \) additional operations over \( F \). Furthermore, if \( |S| \geq 2n \) then the probability of failure of this Las Vegas computation is at most \( k/(2n) \).

Note that a randomized algorithm that terminates with probability one, always returns the desired polynomial \( \text{minpol}(A,v) \), and has an expected complexity as described in Fact 2.8, can be obtained by performing independent trials of a Las Vegas algorithm as described above until an attempt to compute the minimum polynomial succeeds. Since each attempt succeeds with probability at least one-half and the trials are independent, the expected number of trials required before the minimum polynomial is generated is at most two, and the probability that more than \( i \) trials are required is at most \( 1 - 2^i \) for every positive integer \( i \).

Algorithms presented in the sequel will make use of a function \( \text{minpolvec}(A,v,k) \) that receives a matrix \( A \in \mathbb{F}^{n \times n} \), vector \( v \in \mathbb{F}^{n \times 1} \), and a positive integer \( k \) such that \( \text{minpol}(A,v) \) has degree at most \( k \) as input, that terminates with probability one, and returns \( \text{minpol}(A,v) \) as output. It will furthermore be assumed (whenever linearity of expectations is not sufficient to complete a complexity analysis) that this function works by performing independent executions of a function \( \text{minpolvec-1/2}(A,v,k) \), that implements a Las Vegas algorithm with the properties (including worst case complexity) described in Fact 2.2, above, until an execution succeeds.

The techniques of Wiedemann and Kaltofen and Pan can also be used to compute the polynomial \( \text{minpol}(U^T,A,v) \), as can be seen by the following modification of their analysis (see, in particular, Wiedemann [28], pages 60–61). This will be used to prove correctness of algorithms presented in the sequel.

Let \( U \) be an \( A^T \)-invariant subspace of \( \mathbb{F}^{n \times 1} \) with dimension \( d \) for some integer \( d \) such that \( 1 \leq d \leq n \), and let \( X_U \in \mathbb{F}^{n \times d} \) be a matrix with full rank \( d \) whose column space is \( U \); any matrix whose columns form a basis for \( U \) will suffice. This matrix defines a bijection \( \phi \) from \( \mathbb{F}^{d \times 1} \) to \( U \) such that \( \phi(\hat{u}) = X_U \hat{u} \in U \) for every vector \( \hat{u} \in \mathbb{F}^{d \times 1} \). The matrix \( X_U^T \in \mathbb{F}^{d \times n} \) is clearly a matrix with row space \( U^T \).

Now (following Wiedemann), let \( S \subseteq \mathbb{F}^{d \times 1} \) be the span of the vectors \( X_U^Tv, X_U^TAv, X_U^TA^2v, \ldots \) and consider the monic polynomial \( f_{U^TA,v} = x^d + \hat{c}_{-1}x^{d-1} + \cdots + \hat{c}_1x + \hat{c}_0 \) of least degree such that

\[
X_U^TA^i v + \hat{c}_{i-1}X_U^TA^{i-1}v + \cdots + \hat{c}_1X_U^TAv + \hat{c}_0X_U^Tv = 0;
\]

since \( U \) is \( A^T \)-invariant and \( X_U \) has column space \( U \), \( u^T A^i \) is in the row space of \( X_U^T \) for every integer \( i \geq 0 \) and every vector \( u \in U \), and (comparing the above condition with Definition 2.7)

\[
f_{U^T,A,v} = \text{minpol}(U^T,A,v).
\]

Thus, if \( m \) is the degree of \( \text{minpol}(U^T,A,v) \), then the vectors \( X_U^Tv, X_U^TAv, \ldots, X_U^TA^{m-1}v \in \mathbb{F}^{d \times 1} \) are linearly independent and form a basis for \( S \).
The elements of $S$ can be identified with the elements of the ring

$$R = \mathbb{F}[x]/(\text{minpol}(U^T, A, v)).$$

This is accomplished using an $\mathbb{F}$-linear mapping $\hat{\psi}$ from $\mathbb{F}[x]$ to $S$ such that $\hat{\psi}(x^i) = X_U^T A^i v$ for every integer $i \geq 0$. This mapping is clearly surjective and, since the kernel of this mapping is the principle ideal generated by $\text{minpol}(U^T, A, v)$, reduction by this kernel produces a bijection (and, $\mathbb{F}$-linear map) $\psi: R \to S$ as desired.

The set of linear functionals from $R$ to $\mathbb{F}$, $R^*$, may be identified with $R$ by a bijection $\eta: R \to R^*$ defined as follows. Let $m$ be the degree of $\text{minpol}(U^T, A, v)$ as above, and define $\eta(1)$ so that

$$\eta(1)(x^i) = \begin{cases} 0 & \text{if } 0 \leq i \leq m - 2, \\ 1 & \text{if } i = m - 1, \end{cases}$$

and use linearity

$$\eta(1)(cf + g) = c\eta(1)(f) + \eta(1)(g) \quad \text{for all } c \in \mathbb{F} \text{ and } f, g \in R$$

to obtain a definition of $\eta(1)(h)$ for all $h \in R$. A notable property of $\eta(1)$ (which will be used shortly) is that $\eta(1)(x^i h) = 0$ for all $i \geq 0$ if and only if $h = 0$, for any given element $h$ of $R$.

One can next define $\eta(x^i)$, for $1 \leq i \leq m - 1$, by setting $\eta(x^i)(f) = \eta(1)(x^i \cdot f)$ for all $f \in R$, and again using linearity,

$$\eta(cf + g)(h) = c\eta(f)(h) + \eta(g)(h) \quad \text{for } c \in \mathbb{F} \text{ and } f, g, h \in R$$

to uniquely define the linear functional $\eta(f)$ for all $f \in R$ (of course, by specifying its value $\eta(f)(g)$ for all $g \in R$).

It is easy to establish that $\eta$ is an injective $\mathbb{F}$-linear map from $R$ to $R^*$. Since $R$ and $R^*$ have the same dimensions as vector spaces over $\mathbb{F}$, it follows that $\eta$ is a bijection — for if one fixes bases from $R$ and $R^*$ then any $\mathbb{F}$-linear map from $R$ to $R^*$ can be represented by a square matrix, and the fact that $\eta$ is injective implies that the matrix representing $\eta$ is nonsingular.

A dual map $\psi^*: S^* \to R^*$ can be associated with the above bijection $\psi: R \to S$ using the rule

$$\psi^*(l)(g) = l(\psi(g)) \quad \text{for all } l \in S^* \text{ and } g \in R.$$ 

Note that $\psi^*$ is also a bijection — once again, because it is an injective $\mathbb{F}$-linear map between two vector spaces with the same dimension over $\mathbb{F}$.

Now, if $u$ is any element of $U$ then then there exists a unique vector $\hat{u} \in \mathbb{F}^{d \times 1}$ such that

$$\phi(\hat{u}) = X_U \hat{u} = u$$

for the map $\phi$ and matrix $X_U$ described above. A corresponding element $\zeta(u)$ of $S^*$ can be defined as a dot product with $\hat{u}$:

$$\zeta(u)(s) = \hat{u}^T \cdot s \in \mathbb{F} \quad \text{for all } s \in S.$$ 

The resulting map $\zeta: U \to S^*$ is $\mathbb{F}$-linear and clearly surjective, so that it is bijective as well.

Let $u \in U$ be fixed and suppose $g$ is the unique element of $R$ such that $\psi^*(\zeta(u)) = \eta(g)$; then it can be established by the above definitions and linearity of the relevant maps that

$$\eta(1)(x^ig) = \eta(g)(x^i) = \psi^*(\zeta(u))(x^i) = (\zeta(u))(\psi(x^i)) = \zeta(u)(X_U^T A^i v) = \hat{u}^T X_U^T A^i v = u^T A^i v.$$
for every integer \( i \geq 0 \), so that the sequences

\[
u^T v, u^T Av, u^T A^2 v, u^T A^3 v, \ldots \quad \text{and} \quad \eta(1)(g), \eta(1)(xg), \eta(1)(x^2 g), \eta(1)(x^3 g), \ldots \]

are the same. The minimum polynomial of the former sequence is \( \text{minpol}(u^T, A, v) \), by definition, while the minimum polynomial of the latter sequence is

\[
h = \frac{\text{minpol}(U^T, A, v)}{\gcd(\text{minpol}(U^T, A, v), g)}
\]

because this is the monic polynomial \( h \) of least degree such that \( h x^i g = 0 \) in the quotient ring \( R = \mathbb{F}[x]/(\text{minpol}(U^T, A, v)) \) for every integer \( i \geq 0 \). In fact, it is the monic polynomial of least degree such that \( h g = 0 \) in \( R \).

The above relationships hold whether the field \( \mathbb{F} \) is finite or not. Suppose now that \( \mathbb{F} \) is finite, so that \( U \) and \( R \) are as well. In this case, it follows from the above that the probability that \( \text{minpol}(u^T, A, v) = \text{minpol}(U^T, A, v) \), for a uniformly and randomly selected element \( u \) of \( U \), is equal to the probability that a uniformly and randomly selected element of \( R \) is a unit. As indicated by Wiedemann, if \( \mathbb{F} \) is finite with size \( q \) then this probability can be calculated by the formula

\[
\Phi(\text{minpol}(U^T, A, v)) = \prod_{f_i \mid \text{minpol}(U^T, A, v)} \left( 1 - q^{-\deg(f_i)} \right)
\]

where each irreducible factor \( f_i \) of \( \text{minpol}(U^T, A, v) \) appears exactly once in the above summation, and the probability that

\[
\text{minpol}(U^T, A, v) = \text{lcm}_{1 \leq i \leq k} \text{minpol}(u_i^T, A, v)
\]

for uniformly and independently chosen elements \( u_1, u_2, \ldots, u_k \in U \) is given by the formula

\[
\Phi_k(\text{minpol}(U^T, A, v)) = \prod_{f_i \mid \text{minpol}(U^T, A, v)} \left( 1 - q^{-k \deg(f_i)} \right)
\]

Now, as noted by Wiedemann, if \( k > 1 \) then

\[
\Phi_k(\text{minpol}(U^T, A, v)) \geq 1 - \sum_{f_i \mid \text{minpol}(U^T, A, v)} q^{-k \deg(f_i)}
\]

\[
\geq 1 - \left( q^k q^{-k} + q^2 q^{-2k} + q^3 q^{-3k} + \ldots \right)
\]

\[
= 1 - \ln \left( \frac{q^{k-1}}{q^{k-1}} \right)
\]

where the middle inequality is derived using the fact that there are at most \( q^h / h \) monic, irreducible polynomials in \( \mathbb{F}[x] \) with degree \( h \) if \( \mathbb{F} \) is a finite field of size \( q \) (see, for example, Lidl and Niederreiter [18]). As Wiedemann notes, even for \( k = 2 \), this is more than 0.3, so that the desired minimum polynomial is obtained after examining the minimum polynomials of two sequences with probability at least 30%. The probability is more than 70% if \( k = 3 \) and three sequences are considered, and (as seen by evaluating the above function using various values for \( k \)) the probability exceeds 85% when considering four sequences, 90% considering five sequences, and 99% when considering eight.
If $F$ is infinite then of course the above analysis is inapplicable. However, equation (5) and the observations preceding it imply the existence of an element $\hat{u}$ of $U$ such that
\[
\text{minpol}(\hat{u}^T, A, v) = \text{minpol}(U^T, A, v);
\]
indeed, it suffices to choose $\hat{u}$ as the element of $U$ so that $g = 1$, if $g$ is the unique element of $R$ such that $\psi^*(\zeta(\hat{u})) = \eta(g)$ as discussed above.

Suppose now that $u_1, u_2, \ldots, u_l$ is a spanning set for $U$, let $S$ be a finite subset of $F$, and suppose $u$ is chosen as $u = \alpha_1 u_1 + \alpha_2 u_2 + \cdots + \alpha_l u_l$, where the values $\alpha_1, \alpha_2, \ldots, \alpha_l$ are chosen uniformly and independently from $S$. In this case, a slight simplification of an argument used by Kaltofen and Pan [14] (taking advantage of the fact that the vector $v$ is fixed, and not randomly selected along with $u$) implies that
\[
\text{Prob}(\text{minpol}(u^T, A, v) = \text{minpol}(U^T, A, v)) \geq 1 - \frac{\deg(\text{minpol}(U^T, A, v))}{|S|}
\]
(see, in particular, their Lemmas 1 and 2).

Thus minpol($U^T, A, v$) can be computed as the least common multiple of a constant number of polynomials minpol($u^T, A, v$) for uniformly and independently selected vectors $u \in U$ if $F$ is finite, or as minpol($u^T, A, v$) for a single (properly selected) element $u$ of $U$ when $F$ is infinite.

Now, since minpol($A, v$) = minpol($F^{1 \times n}, A, v$), Fact 2.8 can be established from the above analysis by setting $U = F^{n \times 1}$, $d = n$, and setting the matrix $X_U \in F^{n \times n}$ to be the identity matrix; indeed, Wiedemann’s analysis for the finite field case, and Kaltofen and Pan’s for the large field case, can be obtained by making these minor specializations. An algorithm to compute minpol($U^T, A, v$) could be obtained by modifying Wiedemann’s and Kaltofen and Pan’s algorithms for minpol($A, v$), by choosing vectors $u \in U$ (and computing minpol($u^T, A, v$) as before) instead of choosing the vectors “randomly” from $F^{n \times 1}$; the complexity of the resulting algorithm would be bounded by the sum of the corresponding cost mentioned in Fact 2.8 and the cost of selecting a constant number of “random” elements of $U$.

2.3 Minimum Polynomials of Matrices and Subspaces

Definition 2.9. Let $U, V \subseteq F^{n \times 1}$ be $A^T$- and $A$-invariant, respectively, for a matrix $A \in F^{n \times n}$. The minimum polynomial of $U^T$, $A$, and $V$, denoted minpol($U^T, A, V$) is the monic polynomial
\[
x^m + c_{m-1}x^{m-1} + \cdots + c_1 x + c_0
\]
of least degree such that
\[
u^T A^m v + c_{m-1}u^T A^{m-1} v + \cdots + c_1 u^T A v + c_0 u^T v = 0 \quad \text{for all } u \in U \text{ and } v \in V.
\]

Since $V$ is $A$-invariant, $A^i v \in V$ whenever $v \in V$ and $i \geq 0$. It therefore follows, much as before, that $f = \text{minpol}(U^T, A, V)$ if and only if $f$ is the monic polynomial of least degree such that
\[
u^T A^{m+i} v + c_{m-1} u^T A^{m-1+i} v + \cdots + c_1 u^T A^{i+1} v + c_0 u^T A^i v = 0
\]
for all $u \in U$, $v \in V$, and for every integer $i \geq 0$ — and, by linearity, that $f = \text{minpol}(U^T, A, V)$ is also the monic polynomial of least degree such that $u^T s(A) f(A) v = 0$ for every element $u$ of $U$, every element $v$ of $V$, and every polynomial $s \in F[x]$. 10
Definition 2.10. Subspaces $U,V \subseteq F^{n \times 1}$ are $A$-complementary if $U$ is $A^T$-invariant, $V$ is $A$-invariant, and
\[ \minpol(A^T, U) = \minpol(U^T, A) = \minpol(A, V). \]

This is not generally a symmetric relationship between subspaces $U$ and $V$ unless $A$ is a symmetric matrix.

The following example will be important in the sequel: Suppose $u,v \in F^{n \times 1}$ and $f \in F[x]$ such that
\[ \minpol(A^T, u) = \minpol(u^T, A,v) = \minpol(A,v) = f. \]
Let $U = K_{u,v}^{(L)}$ and $V = K_{u,v}^{(R)}$, as defined in Section 2.1. Then, since the above three minimum polynomials are equal, $U$ is an $A^T$-invariant subspace containing $u$, $V$ is an $A$-invariant subspace containing $v$, and, furthermore,
\[ \minpol(A^T, U) = \minpol(U^T, A,V) = \minpol(A,V) = f, \]
so that $U$ and $V$ are $A$-complementary.

As in the previous section, algorithms to compute the above polynomials can be obtained as minor generalizations of algorithms that have been given by Wiedemann [28], for computations over finite fields, and by Kaltofen and Pan [14], for computations over large fields. The next few lemmas will be needed to generalize their analyses and results, and will also be of use in the sequel.

Lemma 2.11. Let $A \in F^{n \times n}$ and let $u,v \in F^{n \times 1}$. If $f = \minpol(u^T, A,v)$ and $g$ is a nonzero polynomial in $F[x]$ then
\[ \minpol((g(A^T)u)^T, A,v) = \minpol(u^T, A,g(A)v) = \frac{f}{\gcd(f,g)}. \]
If $U$ is an $A^T$-invariant subspace, $f = \minpol(U^T, A,v)$ and $g$ is a nonzero polynomial in $F[x]$, then
\[ \minpol(U^T, A,g(A)v) = \frac{f}{\gcd(f,g)}. \]

Proof. Let $g^{(1)} = \gcd(f,g)$, $f^{(2)} = f/g^{(1)}$, and $g^{(2)} = g/g^{(1)}$, so that
\[ \gcd(f^{(2)}, g^{(2)}) = \gcd(f, \gcd(f,g), g/\gcd(f,g)) = 1, \]
and let $h = \minpol((g(A^T)u)^T, A,v)$.

To prove the first claim, recall that since $f = \minpol(u,A,v)$, $u^T f(A)s(A)v = 0$ for every polynomial $s \in F[x]$. Now, if $i \geq 0$ then
\[ (g(A^T)u)^T f^{(2)}(A)A^i v = u^T g(A) f^{(2)}(A)A^i v \]
\[ = u^T g^{(1)}(A)g^{(2)}(A)f^{(2)}(A)v \]
\[ = u^T g^{(2)}(A)f(A)A^i v \]
\[ = u^T f(A)s(A)v \]
\[ = 0, \]
for $s = x^iy^{(2)} \in F[x]$ since $f = \minpol(u^T, A,v)$,

implying that $f^{(2)}$ is divisible by $h$.

Since $f^{(2)}$ and $g^{(2)}$ are relatively prime in $F[x]$, $f^{(2)}/h$ and $g^{(2)}$ are relatively prime in $F[x]$ as well, and there exist polynomials $s,t \in F[x]$ such that $sf^{(2)}/h + tg^{(2)} = 1$. In this case, $sf^{(2)} + tg^{(2)}h = h$.
Now, once again, if \( i \geq 0 \), then one can use a similar derivation to establish that
\[
u^T A^i h(A) g^{(1)}(v) = u^T A^i \left[ s(A) f^{(2)}(A) + t(A) g^{(2)}(A) h(A) \right] g^{(1)}(A) v = u^T A^i s(A) f(A) v + (g(A^T) u)^T A^i t(A) h(A) v = 0,
\]
using the fact that \( f^{(2)} g^{(1)} = f = \minpol(u^T, A, v), g = g^{(1)} g^{(2)}, \) and \( h = \minpol((g(A^T) u)^T, A, v) \). This implies that \( h g^{(1)}(1) \) is divisible by \( f = f^{(2)} g^{(1)}(1) \), and (since \( g^{(1)}(1) \neq 0 \)) that \( h \) is divisible by \( f^{(2)}(2) \).

Since \( h \) and \( f^{(2)} \) are both monic by definition, and each divides the other, \( h = f^{(2)}(2) \) as desired. That is,
\[
\minpol((g(A^T) u)^T, A, v) = \frac{f}{\gcd(f,g)},
\]
Since \( (g(A^T) u)^T s(A) v = u^T s(A) (g(A) v) \) for every polynomial \( s \in \mathbb{F}[x] \) it is easy to establish that
\[
\minpol((g(A^T) u)^T, A, v) = \minpol(u, A, (g(A) v))
\]
for any matrix \( A \in \mathbb{F}^{n \times n} \), pair of vectors \( u, v \in \mathbb{F}^{n \times 1} \), and polynomial \( g \in \mathbb{F}[x] \), completing the proof of the first claim.

The proof of the second claim is almost identical to that of the first: Defining \( g^{(1)}(1), f^{(2)}(2), \) and \( g^{(2)}(2) \) from \( f \) and \( g \) as above, and setting \( h = \minpol(U^T, A, g(A) v) \), one can establish that
\[
u^T f^{(2)}(2) A^i (g(A) v) = (g(A^T) u)^T f^{(2)}(2) A^i v = 0 \quad \text{and} \quad u^T A^i h(A) g^{(1)}(1) v = 0
\]
for every vector \( u \in U \) and integer \( i \geq 0 \), using essentially the derivations given in the proof of the first claim. It can then be argued that \( f^{(2)} \) and \( h \) are monic polynomials dividing one another, so
\[
\minpol(U^T, A, g(A) v) = h = f^{(2)} = \frac{f}{\gcd(f,g)},
\]
as desired.

\[\square\]

**Lemma 2.12.** Let \( A \in \mathbb{F}^{n \times n} \), let \( U \) be an \( A^T \)-invariant subspace of \( \mathbb{F}^{n \times 1} \), let \( V \) be an \( A \)-invariant subspace of \( \mathbb{F}^{n \times 1} \), let \( u_1, u_2 \in U \) and let \( v_1, v_2 \in V \). Suppose that \( f_1 \) and \( f_2 \) are relatively prime polynomials in \( \mathbb{F}[x] \). If
\[
\minpol(U^T, A, v_1) = f_1 \quad \text{and} \quad \minpol(U^T, A, v_2) = f_2
\]
then
\[
\minpol(U^T, A, v_1 + v_2) = f_1 f_2.
\]
Furthermore, if
\[
\minpol(V^T, A^T, u_i) = \minpol(U^T, A, v_i) = f_i
\]
for \( i = 1, 2 \) then
\[
\minpol(V^T, A^T, u_1 + u_2) = \minpol((u_1 + u_2)^T, A, v_1 + v_2) = \minpol(U^T, A, v_1 + v_2) = f_1 f_2.
\]

**Proof.** Consider the first claim, and let \( u \in U \).

Since \( f_1 = \minpol(U^T, A, v_1) \) and \( u \in U \), \( u^T f_1(A) s(A) v_1 = 0 \) for every polynomial \( s \in \mathbb{F}[x] \). In particular (choosing \( s = x^i f_2 \)), \( u^T A^i f_1(A) f_2(A) v_1 = 0 \) for every integer \( i \geq 0 \). Similarly, since \( f_2 = \minpol(U^T, A, v_2) \), \( u^T A^i f_1(A) f_2(A) v_2 = u^T A^i f_2(A) f_1(A) v_2 = 0 \) for every integer \( i \geq 0 \) as well.
Thus $u^T A^i f_1(A) f_2(A)(v_1 + v_2) = u^T A^i f_1(A) f_2(A)v_1 + u^T A^i f_1(A) f_2(A)v_2 = 0$ for every integer $i \geq 0$, implying (since $u$ is arbitrarily chosen from $U$) that $f_1 f_2$ is divisible by $\minpol(U^T, A, v_1 + v_2)$.

Suppose now that $\minpol(U^T, A, v_1 + v_2)$ is a proper divisor of $f_1 f_2$. Then there exists an irreducible polynomial $g \in \mathbb{F}[x]$ and a positive integer $k$ such that $g^k$ divides $f_1 f_2$ but $g^k$ does not divide $\minpol(U^T, A, v_1 + v_2)$.

Since $f_1$ and $f_2$ are relatively prime, either $g^k$ divides $f_1$ or $g^k$ divides $f_2$. Assume, without loss of generality, that $g^k$ divides $f_1$, so that it is relatively prime with $f_2$. Then $g^k$ divides $\minpol(\hat{u}^T, A, v_1)$ for some element $\hat{u}$ of $U$. Consider the polynomials

$$\hat{f}_1 = \minpol(\hat{u}^T, A, v_1 + v_2), \quad \hat{f}_2 = \minpol(\hat{u}^T, A, v_2), \quad \text{and} \quad \hat{f} = \operatorname{lcm}(\hat{f}_1, \hat{f}_2).$$

By an argument similar to the one used at the beginning of this proof, one can establish that $\hat{u}^T A^i \hat{f}(A)(v_1 + v_2) = 0$ for every integer $i \geq 0$, since $\hat{f}$ is divisible by $\hat{f}_1$. Similarly, $\hat{u}^T A^i \hat{f}(A)v_2 = 0$ for every integer $i \geq 0$, since $\hat{f}$ is divisible by $\hat{f}_2$. Therefore

$$\hat{u}^T A^i \hat{f}(A)v_1 = \hat{u}^T A^i \hat{f}(A)((v_1 + v_2) - v_2) = \hat{u}^T A^i \hat{f}(A)(v_1 + v_2) - \hat{u}^T A^i \hat{f}(A)v_2 = 0 - 0 = 0$$

for every integer $i \geq 0$ as well, so that $\hat{f}$ is divisible by $\minpol(\hat{u}^T, A, v_1)$.

On the other hand, $g^k$ divides $\minpol(\hat{u}^T, A, v_1)$ by the choice of $\hat{u}$, so $g^k$ divides $\hat{f}$. However, $g^k$ does not divide $\minpol(\hat{u}^T, A, v_1 + v_2) = \hat{f}_1$, since $g^k$ does not divide $\minpol(U^T, A, v_1 + v_2)$ and $\hat{u} \in U$. It does not divide $\hat{f}_2 = \minpol(\hat{u}^T, A, v_2)$ either, since it does not divide $\minpol(U^T, A, v_2)$. Since $g$ is irreducible and $\hat{f} = \operatorname{lcm}(\hat{f}_1, \hat{f}_2), g^k$ does not divide $\hat{f}$ either, and we have a contradiction. Thus $\minpol(U^T, A, v_1 + v_2)$ and $f_1 f_2$ are associates in $\mathbb{F}[x]$ and, since both polynomials are monic, $\minpol(U^T, A, v_1 + v_2) = f_1 f_2$ as stated in the first claim.

Suppose now that the conditions in the second claim are satisfied, that is,

$$\minpol(V^T, A^T, u_i) = \minpol(u_i^T, A, v_1) = \minpol(U^T, A, v_1) = f_i$$

for $i = 1, 2$. Two applications of the first claim establish that

$$\minpol(V^T, A^T, u_1 + u_2) = \minpol(U^T, A, v_1 + v_2) = f_1 f_2,$$

so it remains only to establish that

$$\minpol((u_1 + u_2)^T, A, v_1 + v_2) = f_1 f_2$$

as well, in order to complete the proof.

Since $\minpol(u_1^T, A, v_2) = \minpol(v_2^T, A^T, u_1)$ divides both $f_1 = \minpol(A^T, u_1)$ and $f_2 = \minpol(A, v_2)$, it also divides $\gcd(f_1, f_2) = 1$. However, it is clear by inspection of Definition 2.1 that this implies that $u_1^T A^T v_2 = 0$ for every integer $i \geq 0$.

A symmetric argument establishes that $u_2^T A^T v_1 = 0$ for every integer $i \geq 0$ as well, so that

$$(u_1 + u_2)^T A^T(v_1 + v_2) = u_1^T A^T v_1 + u_2^T A^T v_2$$

for every integer $i \geq 0$ and, by linearity, that

$$(u_1 + u_2)^T s(A)(v_1 + v_2) = u_1^T s(A)v_1 + u_2^T s(A)v_2$$

for every polynomial $s \in \mathbb{F}[x]$. The remainder of the claim can now be established by proving that $f_1 f_2$ and $\minpol((u_1 + u_2)^T, A, v_1 + v_2)$ are two monic polynomials that divide one another, essentially by repeating the argument used to prove the first claim. □
A similar result will be of use in the sequel.

**Lemma 2.13.** Let $A \in F^{n \times n}$, let $U$ be an $A^T$-invariant subspace of $F^n$, let $V$ be an $A$-invariant subspace of $F^n$, and let $u, v_1, v_2 \in V$. Suppose $f_1$ and $f_2$ are monic polynomials in $F[x]$. If
\[
\minpol(U^T, A, v_1) = f_1 \quad \text{and} \quad \minpol(U^T, A, v_2) = f_2
\]
then $\minpol(U^T, A, v_1 + v_2)$ is divisible by $\lcm(f_1, f_2)$. Furthermore, if
\[
\minpol(U^T, A, v_1) = f_1 \quad \text{and} \quad \minpol(U^T, A, v_2) = f_2
\]
then $\minpol(U^T, A, v_1 + v_2)$ is divisible by $\lcm(f_1, f_2)$ as well.

**Proof.** Let $g = \lcm(f_1, f_2)$. The second claim follows from the fact that, for any polynomial $s \in F[x]$,
\[
u^T s(A)g(A)(v_1 + v_2) = u^T s(A)g(A)v_1 + u^T s(A)g(A)v_2 = u^T s_1(A)f_1(A)v_1 + u^T s_2(A)f_2(A)v_2 = 0 + 0 = 0,
\]
for $s_1 = sg/f_1 \in F[x]$ and $s_2 = sg/f_2 \in F[x]$, and using the fact that $f_i = \minpol(U^T, A, v_i)$ for $i = 1$ and $i = 2$. The first claim follows from the second, and the fact that $\minpol(U^T, A, v_i)$ is a divisor of $\minpol(U^T, A, v_i)$ for $i = 1$ and $i = 2$ and for every element $u$ of $U$. \hfill \Box

**Lemma 2.14.** If $U, V \subseteq F^n$ are $A$-complementary subspaces then there exist vectors $u \in U$ and $v \in V$ such that
\[
\minpol(A^T, u) = \minpol(U^T, A, v) = \minpol(A, v) = \minpol(U^T, A, V).
\]

**Proof.** Suppose
\[
\minpol(U^T, A, V) = \prod_{i=1}^k g_i^{e_i}
\]
for distinct monic irreducible polynomials $g_1, g_2, \ldots, g_k \in F[x]$ and positive integers $e_1, e_2, \ldots, e_k$. Then there must exist elements $\hat{u}_i \in U$ and $\hat{v}_i$ of $V$ such that $\minpol(U_i^T, A, \hat{v}_i)$ is divisible by $g_i^{e_i}$, for $1 \leq i \leq k$.

Let $l_i = \minpol(A^T, \hat{u}_i) = \minpol(F^{1 \times n}, A^T, \hat{u}_i)$. Since $l_i$ is divisible by $\minpol(U_i^T, A, \hat{v}_i)$, $l_i$ is divisible by $g_i^{e_i}$. On the other hand, since $U$ and $V$ are $A$-complementary and $l_i$ divides $\minpol(A^T, U) = \minpol(U^T, A, V)$, $l_i$ is not divisible by $g_i^{e_i+1}$. Now set $\hat{l}_i = l_i/g_i^{e_i}$, so that $\hat{l}_i \in F[x]$ and $\gcd(\hat{l}_i, g_i^{e_i}) = 1$, and set
\[
u_i = \hat{l}_i(A^T)\hat{u}_i;
\]
then $\nu_i \in U$ since $U$ is $A^T$-invariant and $\hat{u}_i \in U$. Furthermore, Lemma 2.11 implies that
\[
\minpol(A^T, u_i) = \minpol(F^{1 \times n}, A^T, \hat{l}_i(A^T)\hat{u}_i) = \frac{l_i}{\gcd(\hat{l}_i, l_i)} = l_i/\hat{l}_i = g_i^{e_i},
\]
and that
\[
\minpol(U_i^T, A, \hat{v}_i) = \minpol((\hat{l}_i(A^T)\hat{u}_i)^T, A, \hat{v}_i) = \frac{\minpol(U_i^T, A, \hat{v}_i)}{\gcd(\minpol(U_i^T, A, \hat{v}_i), l_i)} = g_i^{e_i}
\]
as well, since $\gcd(\minpol(U_i^T, A, \hat{v}_i), \hat{l}_i) = \minpol(U_i^T, A, \hat{v}_i)/g_i^{e_i}$ by the choice of $\hat{u}_i$, $\hat{v}_i$, and $\hat{l}_i$. 14
Next, set \( r_i = \text{minpol}(A, \hat{v}_i) = \text{minpol}(F^{1 \times n}, A, \hat{v}_i) \); \( r_i \) is divisible by \( g_i^{e_i} \) but not by \( g_i^{e_i+1} \), by the same argument as used to prove this for \( l_i \). Set \( \tilde{r}_i = r_i/g_i^{e_i} \), so that \( \tilde{r}_i \in F[x] \) and \( \gcd(\tilde{r}_i, g_i^{e_i}) = 1 \), and set

\[
v_i = \tilde{r}_i(A)\hat{v}_i;
\]

then \( r_i \in V \) since \( V \) is \( A \)-invariant and \( \hat{v}_i \in V \). Now Lemma 2.11 implies that

\[
\text{minpol}(A, v_i) = \text{minpol}(F^{1 \times n}, A, \tilde{r}_i(A)\hat{v}_i) = \frac{r_i}{\gcd(r_i, r_i)} = r_i/\tilde{r}_i = g_i^{e_i},
\]

and that

\[
\text{minpol}(u_i^T, A, v_i) = \text{minpol}(u_i^T, A, \tilde{r}_i(A)\hat{v}_i) = g_i^{e_i}/\gcd(g_i^{e_i}, \tilde{r}_i) = g_i^{e_i},
\]

since \( \text{minpol}(u_i^T, A, \hat{v}_i) = g_i^{e_i} \) and since \( g_i^{e_i} \) and \( \tilde{r}_i \) are relatively prime.

Thus \( u_i \in U, \ v_i \in V, \) and

\[
\text{minpol}(A^T, u_i) = \text{minpol}(u_i^T, A, v_i) = \text{minpol}(A, v_i) = g_i^{e_i}
\]

for \( 1 \leq i \leq k \). Since the polynomials \( g_1^{e_1}, g_2^{e_2}, \ldots, g_k^{e_k} \) are pairwise relatively prime, a repeated application of Lemma 2.12 now suffices to prove that if

\[
u = \sum_{i=1}^{k} u_i \in U \quad \text{and} \quad v = \sum_{i=1}^{k} v_i \in V,
\]

then

\[
\text{minpol}(A^T, u) = \text{minpol}(u^T, A, v) = \text{minpol}(A, v) = \prod_{i=1}^{k} g_i^{e_i} = \text{minpol}(U^T, A, V),
\]

as desired. \( \Box \)

At this point, Wiedemann’s and Kaltofen and Pan’s results can be generalized to obtain efficient algorithms to compute the minimum polynomials of matrices and subspaces (see Proposition 4 of Wiedemann [28], and Lemmas 1 and 2 of Kaltofen and Pan [14]).

In particular, if \( F \) is finite and \( U \) and \( V \) are \( A \)-complementary subspaces of \( F^{n \times 1} \), for a matrix \( A \in F^{n \times n} \), then Lemma 2.14 implies that there exists vectors \( u \in U \) and \( v \in V \) such that

\[
\text{minpol}(u^T, A, v) = \text{minpol}(A^T, u) = \text{minpol}(U^T, A, V).
\]

The generalization of Wiedemann’s analysis given in Subsection 2.2 implies that if \( k > 1 \) and \( k \) vectors \( v_1, v_2, \ldots, v_k \) are selected uniformly and independently from \( V \) then the probability that

\[
\text{minpol}(V^T, A^T, u) = \text{lcm} \minpol(v_i^T, A^T, u)
\]

is

\[
\Phi_k(\text{minpol}(V^T, A^T, u)) \geq 1 - \ln \left( \frac{q^{k-1}}{q^2-1} \right)
\]

where \( |F| = q \) and \( \Phi_k \) is the polynomial introduced in Subsection 2.2. As noted there, this probability exceeds 30% if \( k = 2 \), 50% if \( k = 3 \), and is more than 90% if \( k = 5 \).

Now, since \( \text{minpol}(u^T, A, v) = \text{minpol}(v^T, A^T, u) \),

\[
\text{minpol}(V^T, A^T, u) = \text{minpol}(V^T, A^T, U) = \text{minpol}(U^T, A, V),
\]
because minpol\((V^T, A^T, u)\) is a monic polynomial that is a factor of minpol\((V^T, A^T, U)\) and divisible by minpol\((v^T, A^T, u)\).

Since each polynomial minpol\((v_i^T, A^T, u)\) divides minpol\((v_i^T, A^T, U) = \text{minpol}(U^T, A, v_i)\), and since minpol\((U^T, A, v_i)\) clearly divides minpol\((U^T, A, V)\), it follows that

\[
\text{minpol}(U^T, A, V) = \text{lcm}_{1 \leq i \leq k} \text{minpol}(U^T, A, v_i)
\]

with probability at least \(\Phi_k(\text{minpol}(V^T, A^T, u))\), as well.

Each of the above polynomials \(\text{minpol}(U^T, A, v_i)\) can be computed as the least common multiple of polynomials \(\text{minpol}(u_{i,j}^T, A, v_i)\), for a constant number of uniformly and independently selected vectors \(u_{i,j} \in U\), using a Las Vegas algorithm as described in Subsection 2.2. One can compute each of these minimum polynomials with certainty, using an expected number of operations as described there, by performing independent trials of the algorithm until one succeeds.

Therefore, since \(\Phi_k(\text{minpol}(V^T, A^T, u)) \geq 1/2\) if \(k \geq 3\), it suffices to uniformly and independently select three vectors \(v_1, v_2, v_3\) from \(V\), apply a Las Vegas algorithm to compute the polynomial \(\text{minpol}(U^T, A, v_i)\) corresponding to each, and return their least common multiple as output. The resulting Monte Carlo algorithm uses asymptotically the (expected) number of operations as mentioned in Fact 2.8, plus the cost of computing a constant number of “random” vectors from \(U\) and \(V\), and it either returns the desired minimum polynomial (with probability at least one-half) or a proper factor of it.

Lemma 2.14 and Lemmas 1 and 2 of Kaltofen and Pan [14] can also be combined to produce a Monte Carlo algorithm to compute \(\text{minpol}(U^T, A, V)\) with this complexity, in the large field case.

As it happens, we will need to compute additional values along with \(\text{minpol}(U^T, A, V)\) as part of an algorithm for the Frobenius decomposition of \(A\). Therefore it will be necessary to modify Wiedemann and Kaltofen and Pan’s algorithms, and their analysis, a bit more. This will be discussed (and results will be stated more precisely) in the sequel.

3 Recovery of a Block of the Frobenius Form

The methods and results from Section 2 can now be applied in order to recover a block of the Frobenius form of a matrix. Section 3.1 introduces a method to compute the minimum polynomial \(\text{minpol}(U^T, A, V)\) for a given matrix \(A\) and \(A\)-complementary subspaces \(U\) and \(V\) along with associated values, and Section 3.2 establishes that the Frobenius form of a matrix \(A\) can be obtained by applying this method to progressively smaller subspaces.

3.1 Computing the Minimum Polynomial and Cyclic Vectors of Subspaces

Once again let \(A \in \mathbb{F}^{n \times n}\) and suppose \(U\) and \(V\) are \(A\)-complementary subspaces of \(\mathbb{F}^{n \times 1}\). By Lemma 2.14, there exist vectors \(u \in U\) and \(v \in V\) such that

\[
\text{minpol}(A^T, u) = \text{minpol}(u^T, A, v) = \text{minpol}(A, v) = \text{minpol}(U^T, A, V),
\]

and the definition of “\(A\)-complementary subspaces” (Definition 2.10) implies that this polynomial is also equal to \(\text{minpol}(A^T, U)\) and \(\text{minpol}(A, V)\) as well. It will be shown in this section that one can find such vectors \(u\) and \(v\) efficiently, along dual bases for \(A, u,\) and \(v\) (cf. Definition 2.3). In particular, a Monte Carlo algorithm will be presented that generates vectors \(u \in U\) and \(v \in V\) (along with dual bases for \(A, u,\) and \(v\)) such that

\[
\text{minpol}(A^T, u) = \text{minpol}(u^T, A, v) = \text{minpol}(A, v),
\]
function filterp \((f, g)\)

begin function
\[
d := g
\]
\[
h := f
\]
while \(\text{deg}(d) > 0\) do
\[
h := h/d
\]
\[
d := \gcd(h, d)
\]
end while
return \(h\)
end function

Figure 1: Function filterp

and such that \(\text{minpol}(u^T, A, v) = \text{minpol}(U^T, A, V)\) with probability at least one-half. Since \(u \in U\) \(v \in V\), \(\text{minpol}(u^T, A, v)\) will be clearly be a factor of \(\text{minpol}(U^T, A, V)\) in any case.

To begin, let us suppose that we are given vectors \(\hat{u} \in U\) and \(\hat{v} \in V\), and that we wish to find vectors \(u \in K_{\hat{u}, \hat{v}}(L) \subseteq U\) and \(v \in K_{\hat{u}, \hat{v}}(R) \subseteq V\) such that
\[
\text{minpol}(A^T, u) = \text{minpol}(u^T, A, v) = \text{minpol}(A, v)
\]
as above, but also such that the degree of \(\text{minpol}(u^T, A, v)\) is kept high. More precisely, if \(g \in \mathbb{F}[x]\) is an irreducible polynomial and \(k\) is a positive integer such that \(g^k\) divides \(\text{minpol}(\hat{u}^T, A, \hat{v})\), but \(g^{k+1}\) does not divide either one of \(\text{minpol}(A^T, \hat{u})\) or \(\text{minpol}(A, \hat{v})\), then we will require that \(g^k\) divide \(\text{minpol}(u^T, A, v)\).

Consider the function \(\text{filterp}\) shown in Figure 1, assuming that its inputs are monic polynomials \(f, g \in \mathbb{F}[x]\) such that \(g\) divides \(f\).

Lemma 3.1. Given monic polynomials \(f, g \in \mathbb{F}[x]\) such that \(g\) divides \(f\), function \(\text{filterp}\) returns the monic polynomial \(h\) of greatest degree such that \(h\) divides \(f\) and \(\gcd(g, h) = 1\).

If the degree of \(f\) is at most \(m\) then the function can be implemented as a deterministic algorithm that uses \(O(m^2)\) operations in \(\mathbb{F}\), with standard polynomial arithmetic.

Proof. Consider the polynomials \(d\) and \(h\) maintained by this algorithm. It is clear by inspection of the algorithm that if \(s \in \mathbb{F}[x]\) is any polynomial that divides \(f\) such that \(\gcd(s, g) = 1\) then \(s\) divides \(h\) before the first execution of the \textbf{while} loop and, since \(d\) is always divisible by \(g\), \(s\) divides \(h\) after each execution of this loop body as well. Inspection of the code should also confirm that \(h\) is always monic, and that if \(t\) is any irreducible polynomial dividing \(g\), then \(t\) divides \(h\) if and only if \(t\) divides \(d\), both before the first execution of the loop body and after each execution of it.

Inspection of the code confirms, as well, that the degree of the polynomial \(h\) decreases by at least one, on each execution of the loop body, and that \(d\) is always a divisor of \(h\). Consequently the degree of \(d\) is zero and the algorithm terminates after at most \(m\) executions of the loop body. At this point, the above loop invariants imply that \(h\) is the monic polynomial of greatest degree dividing \(f\) such that \(\gcd(g, h) = 1\), as desired.

Let \(h_0 = f\) and \(d_0 = g\), so that \(h_0\) and \(d_0\) are the values of \(h\) and \(d\), respectively, at the beginning of the first execution of the loop body. Suppose the loop is executed \(k\) times and let
function filterv \((A, \hat{u}, \hat{v}, k)\)

begin function

\[f_m := \text{minpolseq}(\hat{u}^T, A, \hat{v}, k)\]
\[f_l := \text{minpolvec}(A^T, \hat{u}, k)\]
\[f_r := \text{minpolvec}(A, \hat{v}, k)\]
\[f_m := \text{filterp}(f_m, \gcd(f_m, f_l/f_m))\]
\[f := \text{filterp}(f_m, \gcd(f_m, f_r/f_m))\]
\[g_l := f_l/f\]
\[g_r := f_r/f\]
\[u := g_l(A^T)\hat{u}\]
\[v := g_r(A)\hat{v}\]

return \(u, v, f\)

end function

Figure 2: Function filterv

Let \(h_i\) and \(d_i\) be the values of \(h\) and \(d\), respectively, at the end of the \(i\)th execution of the loop body, for \(1 \leq i \leq k\). Since \(h_i = h_{i-1}/d_{i-1}\) for \(1 \leq i \leq k\) and \(d_k = 1\) it is clear that \(f = h_0\) is divisible by \(\prod_{i=0}^{k} d_i\). Thus, if \(d_i\) has degree \(m_i\) for \(0 \leq i \leq k\) then \(\sum_{i=0}^{k} m_i \leq m\). Now, a careful analysis of the cost of both polynomial division with remainder and computation of the greatest common divisor of polynomials, using standard arithmetic, confirms that each operation can be performed using \(O(mm_r)\) operations over \(\mathbb{F}\) on inputs \(f_l\) and \(f_r\) with degrees \(m_l\) and \(m_r\) respectively (see, for example, von zur Gathen and Gerhard [10]). It can therefore be established that the \(i\)th execution of the loop body can be executed, with standard polynomial arithmetic, using \(O(m^2m_{r-1})\) operations for \(1 \leq i \leq k\), and that the entire function can be executed using \(O(m^2)\) operations over \(\mathbb{F}\) as claimed. \(\square\)

Now consider the function filterv shown in Figure 2.

**Lemma 3.2.** Let \(A \in \mathbb{F}^{n \times n}\) and let \(U\) and \(V\) be \(A\)-complementary subspaces. Let \(\hat{u} \in U\), \(\hat{v} \in V\), and let \(k\) be an integer such that the degrees of both \(\minpol(A^T, \hat{u})\) and \(\minpol(A, \hat{v})\) are both at most \(k\). Then, given the inputs \(A\), \(\hat{u}\), \(\hat{v}\) and \(k\), the function filterv returns vectors \(u \in U\), \(v \in V\), and a polynomial \(f \in \mathbb{F}[x]\) such that

\[\minpol(A^T, u) = \minpol(u^T, A, v) = \minpol(A, v) = f\]

and such that, for every irreducible polynomial \(g \in \mathbb{F}[x]\) and nonnegative integer \(m\), if \(g^m\) divides \(\minpol(\hat{u}^T, A, \hat{v})\) but \(g^{m+1}\) does not divide either \(\minpol(A^T, \hat{u})\) or \(\minpol(A, \hat{v})\), then \(g^m\) divides \(f\).

The function can be implemented as a randomized algorithm that terminates with probability one, so that it selects vectors whose entries are selected uniformly and independently from \(\mathbb{F}\) if \(\mathbb{F}\) is finite, or from a finite subset \(S\) of size at least \(2n\) if \(\mathbb{F}\) is infinite. The expected number of vectors selected by this algorithm is in \(O(1)\). The expected number of matrix-times-vectors products (using matrix \(A^T\) or \(A\)) computed by this algorithm is in \(O(1)\), and the expected number of additional operations over \(\mathbb{F}\) performed by this algorithm is in \(O(kn)\), using standard polynomial arithmetic.

**Proof.** It is clear by inspection of the code in Figure 2 (and a review of the specifications of functions minpolseq and minpolvec, from Section 2) that the polynomials \(f_m\), \(f_l\) and \(f_r\) have initial
function mergev \((A, u_1, v_1, f_1, u_2, v_2, f_2)\)

begin function

\[
g_1 := \text{filter}(f_1, \text{lcm}(f_1, f_2)/f_2)
\]

\[
h_1 := f_1/g_1
\]

\[
h_2 := \text{filter}(f_2, \text{gcd}(h_1, f_2))
\]

\[
g_2 := f_2/h_2
\]

\[
u := g_1(A^Tu_1 + g_2(A^Tu_2)
\]

\[
v := g_1(A)v_1 + g_2(A)v_2
\]

\[
f := h_1 \cdot h_2
\]

return \(u, v, f\)

end function

Figure 3: Function mergev

values \(\text{minpol}(\hat{u}^T, A, \hat{v}), \text{minpol}(A^T, \hat{u}), \text{and minpol}(A, \hat{v})\) respectively. Lemma 3.1 therefore implies that, immediately after the first application of filterp, \(f_m\) is the monic polynomial of greatest degree that divides \(\text{minpol}(\hat{u}^T, A, \hat{v})\) and is relatively prime to \(\text{minpol}(A^T, \hat{u})/\text{minpol}(\hat{u}^T, A, \hat{v})\). This lemma also implies that, immediately after the second application of this function, \(f\) is the monic polynomial of greatest degree that divides \(\text{minpol}(\hat{u}^T, A, \hat{v})\) and is relatively prime to both \(\text{minpol}(A^T, \hat{u})/\text{minpol}(\hat{u}^T, A, \hat{v})\) and \(\text{minpol}(A, \hat{v})/\text{minpol}(\hat{u}^T, A, \hat{v})\). It follows that if \(u \in K^{(L)}_{u,v}\) and \(v \in K^{(R)}_{u,v}\) are vectors such that

\[
\text{minpol}(A^T, u) = \text{minpol}(u^T, A, v) = \text{minpol}(A, v) = f,
\]

then \(u, v,\) and \(f\) are correct outputs. Correctness of the algorithm now follows by inspection of the rest of the code and by application of Lemma 2.11.

It should next be noted that if \(k\) is the given bound on the degrees of \(\text{minpol}(A^T, \hat{u})\) and \(\text{minpol}(A, \hat{v})\), then \(k\) also bounds the degrees of all values assumed by \(f_m\) and \(f\), as well as the polynomials \(f_1, f_r, g_1\), and \(g_r\). The stated complexity bounds now follow by applications of Fact 2.2 and the specification of function minpolseq, Fact 2.8 and the specification of function minpolvec, Lemma 3.1, and an inspection of the code.

Function mergev is shown in Figure 3 and will also be required.

**Lemma 3.3.** Suppose again that \(A \in \mathbb{F}^{n \times n}\) and that \(U\) and \(V\) are \(A\)-complementary subspaces. Let \(u_1, u_2 \in U, v_1, v_2 \in V,\) and let \(f_1, f_2 \in \mathbb{F}[x]\) such that

\[
\text{minpol}(A^T, u_i) = \text{minpol}(u_i^T, A, v_i) = \text{minpol}(A, v_i) = f_i
\]

for \(i = 1, 2\). Then, given \(A, u_1, v_1, f_1, u_2, v_2,\) and \(f_2\) as input, the function mergev returns vectors \(u \in U\) and \(v \in V\) and a polynomial \(f \in \mathbb{F}[x]\) such that

\[
\text{minpol}(A^T, u) = \text{minpol}(u^T, A, v) = \text{minpol}(A, v) = f = \text{lcm}(f_1, f_2).
\]

If \(1 \leq k \leq n\) and the degree of \(\text{minpol}(U^T, A, V)\) is at most \(k\), then this function can be implemented as a deterministic algorithm that uses \(O(k)\) multiplications of \(A^T\) by vectors in \(U\), \(O(k)\) multiplications of \(A\) by vectors in \(V\), and \(O(n + k^2)\) additional arithmetic operations over \(\mathbb{F}\), with standard polynomial arithmetic.
Proof. Suppose
\[ f_1 = \prod_{i=1}^{k} \hat{g}_i^{a_i} \quad \text{and} \quad f_2 = \prod_{i=1}^{k} \hat{g}_i^{b_i} \]
for distinct monic, irreducible polynomials \( \hat{g}_1, \hat{g}_2, \ldots, \hat{g}_k \), and consider the polynomials \( g_1, g_2, h_1, h_2 \), and \( f \) that are generated by this function. Clearly
\[ \text{lcm}(f_1, f_2) = \prod_{i=1}^{k} \hat{g}_i^{\max(a_i, b_i)}, \]
so that
\[ \text{lcm}(f_1, f_2)/f_2 = \prod_{i=1}^{k} \hat{g}_i^{\max(a_i, b_i) - b_i} \]
is divisible by \( f_1 f_2 / f_2 = f_1 \) and, by Lemma 3.1,
\[ g_1 = \text{filterp}(f_1, \text{lcm}(f_1, f_2)/f_2) = \prod_{i=1}^{k} \hat{g}_i^{c_i}, \quad \text{for} \quad c_i = \begin{cases} a_i & \text{if} \ a_i \leq b_i, \\ 0 & \text{if} \ a_i > b_i. \end{cases} \]
Therefore
\[ h_1 = f_1/g_1 = \prod_{i=1}^{k} \hat{g}_i^{d_i}, \quad \text{for} \quad d_i = \begin{cases} 0 & \text{if} \ a_i \leq b_i, \\ a_i & \text{if} \ a_i > b_i. \end{cases} \]
Since \( f_1 \) is divisible by \( g_1, h_1 = f_1/g_1 \), and \( g_1 \) and \( h_1 \) are relatively prime, three applications of Lemma 2.11 from Section 2.3 can be used to establish that
\[ \text{minpol}(A^T, g_1(A^T)u_1) = \text{minpol}((g_1(A^T)u_1)^T, A, g_1(A)v_1) = \text{minpol}(A, g_1(A)v_1) = h_1. \quad (10) \]
Lemma 3.1 and the above factorization of \( h_1 \) imply that
\[ h_2 = \text{filterp}(f_2, \text{gcd}(h_1, f_2)) = \prod_{i=1}^{k} \hat{g}_i^{e_i}, \quad \text{for} \quad e_i = \begin{cases} b_i & \text{if} \ a_i \leq b_i, \\ 0 & \text{if} \ a_i > b_i, \end{cases} \]
so that
\[ g_2 = f_2/h_2 = \prod_{i=1}^{k} \hat{g}_i^{l_i}, \quad \text{for} \quad l_i = \begin{cases} 0 & \text{if} \ a_i \leq b_i, \\ b_i & \text{if} \ a_i > b_i. \end{cases} \]
Since \( f_2 \) is divisible by \( g_2, h_2 = f_2/g_2 \), and \( g_2 \) and \( h_2 \) are relatively prime, another three applications of Lemma 2.11 establish that
\[ \text{minpol}(A^T, g_2(A^T)u_2) = \text{minpol}((g_2(A^T)u_2)^T, A, g_2(A)v_2) = \text{minpol}(A, g_2(A)v_2) = h_2. \quad (11) \]
Since \( h_1 \) and \( h_2 \) are relatively prime, equations (10) and (11), Lemma 2.12, and the definitions of \( u \) and \( v \) in the code now imply that
\[ \text{minpol}(A^T, u) = \text{minpol}(u^T, A, v) = \text{minpol}(A, v) = h_1 h_2. \]
The factorizations of \( h_1 \) and \( h_2 \) given above imply that \( h_1 h_2 = \prod_{i=1}^{k} \hat{g}_i^{\max(a_i, b_i)} = \text{lcm}(f_1, f_2) \), as needed to conclude that the function is correct.

The complexity bounds stated in the lemma can be established by a final application of Lemma 3.1 and an inspection of the code. \( \square \)

20
Suppose, once again, that $F$ is finite with size $q$, that vectors $\hat{u}_1, \hat{u}_2, \ldots$ are chosen uniformly and independently from $U$, and that vectors $\hat{v}_1, \hat{v}_2, \ldots$ are chosen uniformly and independently from $V$. If $g$ is an irreducible polynomial and $m$ is a positive integer such that $g^m$ divides $\text{minpol}(U^T, A, V)$ but $g^{m+1}$ does not, then, since $U$ and $V$ are $A$-complementary, so that $\text{minpol}(U^T, A, V) = \text{minpol}(A, V)$, the probability that $g^m$ divides $\text{minpol}(A, \hat{v}_i)$ is at least

$$1 - q^{-\deg(g)};$$

see the analysis in Section 2.3 for details. Furthermore, the conditional probability that $g^m$ also divides $\text{minpol}(\hat{u}_i^T, A, \hat{v}_i)$ if it divides $\text{minpol}(A, \hat{v}_i)$ is equal to the above probability; see the analysis in Section 2.2 for a justification. It therefore follows that $g^m$ divides $\text{minpol}(\hat{u}_i^T, A, \hat{v}_i)$ with at least the square of the above probability, so that the probability that $g^m$ does not divide this minimum polynomial is at most

$$2q^{-\deg(g)} - q^{-2\deg(g)}.$$

Thus Lemma 3.2 implies that if $u_i \in U$, $v_i \in V$, and $f_i \in F[x]$ are produced by the function $\text{filterv}$ when given $A$, $\hat{u}_i$, $\hat{v}_i$, and an upper bound on $\text{minpol}(U^T, A, V)$ as inputs, then

$$\text{minpol}(A^T, u_i) = \text{minpol}(u_i^T, A, v_i) = \text{minpol}(A, v_i) = f_i,$$

and this polynomial is divisible by $g^m$ with probability at least $1 - 2q^{-\deg(g)} + q^{-2\deg(g)}$. Furthermore, since $\hat{u}_1, \hat{u}_2, \ldots, \hat{u}_l$ and $\hat{v}_1, \hat{v}_2, \ldots, \hat{v}_l$ are independently selected, the probability that $\text{lcm}_{i=1}^l f_i$ is not divisible by $g^m$ is at most

$$\left(2q^{-\deg(g)} - q^{-2\deg(g)}\right)^l$$

and, therefore,

$$\text{lcm}_{1 \leq i \leq l} f_i = \text{minpol}(U^T, A, V)$$

with probability at least

$$\Psi_l(\text{minpol}(U^T, A, V)) = \prod_{g_i | \text{minpol}(U^T, A, V), g_i \text{ is irreducible}} 1 - \left(2q^{-\deg(g_i)} - q^{-2\deg(g_i)}\right)^l.$$

Once again (following Wiedemann’s argument),

$$\Psi_l(\text{minpol}(U^T, A, V)) \geq 1 - \sum_{g_i | \text{minpol}(U^T, A, V), g_i \text{ is irreducible}} \left(2q^{-\deg(g_i)} - q^{-2\deg(g_i)}\right)^l \geq 1 - \sum_{h \geq 1} \frac{q^{h^3}}{h} \left(2q^{-h} - q^{-2h}\right)^l,$$

once again, using the fact that there are at most $q^h/h$ monic irreducible polynomials of degree $h$...
in \( F[x] \). Now if \( q = 2 \) this implies that

\[
\Psi_l(\text{minpol}(U^T, A, V)) \geq 1 - 2 \left( \frac{3}{4} \right)^l - \sum_{h \geq 2} \frac{q^h}{h!} \left( \frac{1}{2} \right)^h
\]

\[
= 1 - 2 \left( \frac{3}{4} \right)^l - 2 \sum_{h \geq 2} \frac{q^{h-1}}{h-1} \left( \frac{1}{2} \right)^h
\]

\[
\geq 1 - 2 \left( \frac{3}{4} \right)^l - 2 \sum_{h \geq 2} \frac{q^{h-1}}{h-1} \left( \frac{1}{2} \right)^h
\]

\[
= 1 - 2 \left( \frac{3}{4} \right)^l - 2 \sum_{j \geq 1} \frac{2^{-j(l-1)}}{j}
\]

\[
= 1 - 2 \left( \frac{3}{4} \right)^l - 2 \ln \left( \frac{2^{-1}}{q^{l-1}} \right)
\]

implying that \( \Psi_l(\text{minpol}(U^T, A, V)) \geq 0.5 \) if \( q = 2 \) and \( l \geq 6 \).

If \( q \geq 3 \), then it follows by the above inequalities that

\[
\Psi_l(\text{minpol}(U^T, A, V)) \geq 1 - \sum_{h \geq 1} \frac{q^h}{h!} \left( \frac{2}{q-1} \right)^h
\]

\[
= 1 - \sum_{h \geq 1} \frac{1}{h!} q^{-h(l-1)}
\]

\[
= 1 - 2 \sum_{h \geq 1} \frac{1}{h!} q^{-h(l-1)}
\]

\[
\geq 1 - 2 \sum_{h \geq 1} \frac{1}{h!} q^{h(l-1)} q^{-h(l-1)}
\]

\[
= 1 - 2 \sum_{h \geq 1} \frac{(q/2)^{-h(l-1)}}{h}
\]

\[
= 1 - 2 \ln \left( \frac{q^{l-1}}{q^{l-2}} \right)
\]

implying that \( \Psi_l(\text{minpol}(U^T, A, V)) \geq 0.5 \) when \( q = 3 \) and \( l \geq 5 \), when \( q = 4 \) and \( l \geq 4 \), when \( 5 \leq q \leq 9 \) and \( l \geq 3 \), and when \( q \geq 11 \) and \( l \geq 2 \).

If \( F \) is a sufficiently large field then it can be argued that

\[
\text{minpol}(A^T, u) = \text{minpol}(u^T, A, v) = \text{minpol}(A, v) = \text{minpol}(U^T, A, V)
\]

for “randomly” chosen elements \( u \) from \( U \) and \( v \) from \( V \), if \( U \) and \( V \) are \( A \)-complementary subspaces.

More precisely, it follows by Lemma 2.14 that vectors \( u \in U \) and \( v \in V \) do exist that satisfy the above condition. Now suppose \( S \) is a finite subset of \( F \) containing at least \( 4k \) distinct elements, where \( k \) is greater than or equal to the degree of minpol \( (U^T, A, V) \). Suppose \( \overline{u}_1, \overline{u}_2, \ldots, \overline{u}_{m_1} \) is a spanning set for \( U \) and that \( \overline{v}_1, \overline{v}_2, \ldots, \overline{v}_{m_2} \) is a spanning set for \( V \). Then if

\[
u = \alpha_1 \overline{u}_1 + \alpha_2 \overline{u}_2 + \cdots + \alpha_{m_1} \overline{u}_{m_1} \in U \quad \text{and} \quad v = \beta_1 \overline{v}_1 + \beta_2 \overline{v}_2 + \cdots + \beta_{m_2} \overline{v}_{m_2} \in V,
\]

where the values \( \alpha_1, \alpha_2, \ldots, \alpha_{m_1}, \beta_1, \beta_2, \ldots, \beta_{m_2} \) are chosen uniformly and independently from \( S \), then it follows by a trivial modification of the argument given by Kaltofen and Pan to prove their Lemmas 1 and 2 that

\[
\text{minpol}(U^T, A, V) = \text{minpol}(U^T, A, V)
\]
function minpolspace \((A; k; \hat{u}_1, \hat{u}_2, \ldots, \hat{u}_l; \hat{v}_1, \hat{v}_2, \ldots, \hat{v}_l)\)

begin function

\[ u, v, f := \text{filterv}(A, \hat{u}_1, \hat{v}_1, k) \]

for \(i := 2 \ldots \text{trialbound}(k)\) do

\[ u, v, f := \text{mergev}(A, u, v, f, \text{filterv}(A, \hat{u}_i, \hat{v}_i, k)) \]

end for

return \(u, v, f\)

end function

Figure 4: Function minpolspace

with probability at least

\[ 1 - \frac{2 \deg(\minpol(U^T, A, V))}{|\mathcal{S}|} \geq \frac{1}{2}. \]

Now since \(U\) and \(V\) are \(A\)-complementary,

\[ \minpol(A^T, U) = \minpol(U^T, A, V) = \minpol(A, V) \]

and, since \(\minpol(A^T, u)\) (respectively, \(\minpol(A, v)\)) is monic, has \(\minpol(U^T, A, v)\) as a factor and divides \(\minpol(A^T, U)\) (respectively, \(\minpol(A, V)\)), condition (12) would imply that

\[ \minpol(A^T, u) = \minpol(A, v) = \minpol(U^T, A, V) \]

as well.

Now let

\[ l = \text{trialbound}(k) = \begin{cases} 6 & \text{if } |\mathcal{F}| = 2, \\ 5 & \text{if } |\mathcal{F}| = 3, \\ 4 & \text{if } |\mathcal{F}| = 4, \\ 3 & \text{if } 5 \leq |\mathcal{F}| \leq 9, \\ 2 & \text{if } 11 \leq |\mathcal{F}| \leq 4k, \\ 1 & \text{otherwise}, \end{cases} \]

and consider the function minpolspace shown in Figure 4.

The following result is a straightforward consequence of the above analysis, along with the information about functions filterv and mergev given in Lemmas 3.2 and 3.3, respectively.

**Theorem 3.4.** Let \(A \in \mathbb{F}^{n \times n}\) and let \(U\) and \(V\) be \(A\)-complementary subspaces. Let \(k\) be an upper bound on the degree of \(\minpol(U^T, A, V)\), let \(l = \text{trialbound}(k)\), let \(\hat{u}_1, \hat{u}_2, \ldots, \hat{u}_l \in U\), and let \(\hat{v}_1, \hat{v}_2, \ldots, \hat{v}_l \in V\). Then, given inputs \(A, k, \hat{u}_1, \hat{u}_2, \ldots, \hat{u}_l\) and \(\hat{v}_1, \hat{v}_2, \ldots, \hat{v}_l\), the function minpolspace returns vectors \(u \in U\) and \(v \in V\) and a monic polynomial \(f \in \mathbb{F}[x]\) such that

\[ \minpol(A^T, u) = \minpol(u^T, A, v) = \minpol(A, v) = f \]

and \(f\) is a divisor of \(\minpol(U^T, A, V)\). Furthermore, \(f = \minpol(U^T, A, V)\) with probability at least one-half if either
(a) the field $F$ is finite, vectors $\hat{u}_1, \hat{u}_2, \ldots, \hat{u}_t$ are chosen uniformly and independently from $U$, and vectors $\hat{v}_1, \hat{v}_2, \ldots, \hat{v}_t$ are chosen uniformly and independently$^2$ from $V$; or

(b) the field $F$ is infinite, $S$ is a finite subset of $F$ with size at least $4k$,

$$
\hat{u}_i = \alpha_{i,1} \overline{u}_1 + \alpha_{i,2} \overline{u}_2 + \cdots + \alpha_{i,m_1} \overline{u}_{m_1} \quad \text{and} \quad \hat{v}_i = \beta_{i,1} \overline{v}_1 + \beta_{i,2} \overline{v}_2 + \cdots + \beta_{i,m_2} \overline{v}_{m_2}
$$

for $1 \leq i \leq l$, where $\overline{u}_1, \overline{u}_2, \ldots, \overline{u}_{m_1}$ is a spanning set for $U$, $\overline{v}_1, \overline{v}_2, \ldots, \overline{v}_{m_2}$ is a spanning set for $V$, and the coefficients $\alpha_{i,j}$ (for $1 \leq j \leq m_1$) and $\beta_{i,j}$ (for $1 \leq j \leq m_2$) are all chosen uniformly and independently from $S$.

The function can be implemented as a randomized algorithm that terminates with probability one. The expected number of vectors chosen by the algorithm from $U$ and $V$ is in $O(1)$, the expected number of matrix-times-vector products (for matrices $A$ or $A^T$) is in $O(k)$, and the expected number of additional operations performed by the algorithm over $F$ is in $O(kn)$, using standard polynomial arithmetic.

**Note 3.5.** An additional result will be needed in the sequel, for the case that $F$ is finite and (as usual) $U$ and $V$ are $A$-complementary subspaces.

Recall, from Section 2.1, that if $u \in U$ and $v \in V$, then $K_{u,v}^{(L)}$ is the vector space spanned by the vectors

$$
u, (A^T)u, (A^T)u, (A^T)^2u, (A^T)^3u, \ldots
$$

and that $K_{u,v}^{(R)}$ is the vector space spanned by the vectors

$$v, Av, A^2v, A^3v, \ldots
$$

If $U$ and $V$ are $A$-complementary then $K_{u,v}^{(L)} \subseteq U$ and $K_{u,v}^{(R)} \subseteq V$.

Suppose that $m > 0$, $l = \text{trialbound}(k)$, where as usual $k$ is greater than or equal to the degree of $\text{minpol}(U^T, A, V)$, that vectors

$$\hat{u}_{1,1}, \ldots, \hat{u}_{1,l}, \ldots, \hat{u}_{m,1}, \ldots, \hat{u}_{m,l}
$$

are chosen uniformly and independently from $U$, and that

$$\hat{v}_{1,1}, \ldots, \hat{v}_{1,l}, \ldots, \hat{v}_{m,1}, \ldots, \hat{v}_{m,l}
$$

are chosen uniformly and independently from $V$. By Theorem 3.4, if $\overline{u}_1 \in U$ and $\overline{v}_1 \in V$ are the vectors and $\overline{f}_i \in F[x]$ is the polynomial produced by function $\text{minpolspace}$ on inputs $A$, $k$, $\hat{u}_{i,1}, \hat{u}_{i,2}, \ldots, \hat{u}_{i,l}$ and $\hat{v}_{i,1}, \hat{v}_{i,2}, \ldots, \hat{v}_{i,l}$, then

$$\text{minpol}(A^T, \overline{u}_i) = \text{minpol}(\overline{u}_i^T, A, \overline{v}_i) = \text{minpol}(A, \overline{v}_i) = \overline{f}_i,
$$

and $\overline{f}_i = \text{minpol}(U^T, A, V)$ with probability at least one-half. Now let $u_1 = \overline{u}_1$, $v_1 = \overline{v}_1$, and $f_1 = \overline{f}_1$, and let $u_i \in U$, $v_i \in V$, and $f_i \in F[x]$ be the vectors and polynomial produced by the function $\text{mergev}$ on inputs $u_{i-1}, v_{i-1}, f_{i-1}$ and $\overline{u}_i, \overline{v}_i, \overline{f}_i$, for $2 \leq i \leq m$, so that

$$
\begin{align*}
\text{minpol}(A^T, u_i) &= \text{minpol}(u_i, A, v_i) = \text{minpol}(A, v_i) = f_i = \lim_{1 \leq j \leq i} \overline{f}_j
\end{align*}
$$

\[\ldots\] from one another and from $\hat{u}_1, \hat{u}_2, \ldots, \hat{u}_t$. 

24
for $1 \leq i \leq m$. Since $\overline{T}_j$ divides $\minpol(U^T, A, V)$ for $1 \leq j \leq m$, $f_i$ clearly divides $\minpol(U^T, A, V)$ for $1 \leq i \leq m$ as well, and it follows by the independence of the vectors $\hat{u}_{r,s}$ and $\hat{v}_{r,s}$ for $1 \leq r \leq m$ and $1 \leq s \leq l$ that

$$f_i = \minpol(U^T, A, V)$$

with probability at least $1 - 2^{-i}$, for $1 \leq i \leq m$.

However, these vectors and functions will not be used in quite this way in the sequel. In particular, it may happen that several applications of the function $\minpol$ are used in sequence to try to discover $\minpol(U^T, A, V)$ and that the results of these applications are combined using the function $\mergevec$ as described above. The first application of $\minpol$ will involve uniformly and independently selected vectors

$$\hat{u}_{1,1}, \ldots, \hat{u}_{1,l} \in U \quad \text{and} \quad \hat{v}_{1,1}, \ldots, \hat{v}_{1,l} \in V$$

as above. In each later application, say, the $i^{\text{th}}$, the uniformly (and independently) selected vector $\hat{u}_{i,j}$, mentioned above, will be replaced as an input to $\minpol$ with a vector

$$\hat{u}_{i,j}' = \hat{u}_{i,j} + \hat{u}_{i,j,1} + \hat{u}_{i,j,2} + \cdots + \hat{u}_{i,j,i-1}$$

where $\hat{u}_{i,j,h} \in K_{\eta_h, \xi_h} \subseteq U$, for $1 \leq h \leq i - 1$, and where the vectors $\eta_h$ and $\xi_h$ are as above. Similarly, the vector $\hat{v}_{i,j}$ will be replaced by as an input to $\minpol$ with a vector

$$\hat{v}_{i,j}' = \hat{v}_{i,j} + \hat{v}_{i,j,1} + \hat{v}_{i,j,2} + \cdots + \hat{v}_{i,j,i-1}$$

where $\hat{v}_{i,j,h} \in K_{\eta_h, \xi_h}^{(R)} \subseteq V$. Fortunately, Lemma 2.13 and a straightforward (but tedious) proof by induction can be used to prove that the polynomials $f_1, f_2, \ldots, f_m$ one obtains are not changed by this substitution: Note that if $g \in \mathbb{F}[x]$ is irreducible, $r$ is a positive integer such that the polynomial $\minpol(U^T, A, V)$ is divisible by $g^r$ but not by $g^{r+1}$, and none of $f_1, f_2, \ldots, f_{i-1}$ are divisible by $g^r$, then Lemma 2.13 can be applied to prove that $g^r$ divides $\minpol(\hat{u}_{i,j}^T, A, \hat{v}_{i,j})$ if and only if $g^r$ divides $\minpol((\hat{u}_{i,j}')^T, A, \hat{v}_{i,j}')$, for $1 \leq r \leq l$, and for the vectors $\hat{u}_{i,j}, \hat{v}_{i,j}, \hat{u}_{i,j}'$ and $\hat{v}_{i,j}'$ as described above.

Consequently $\minpol(U^T, A, V)$ will be available after $m$ applications of $\minpol$ with probability at least $1 - 2^{-m}$, even if the above replacement of inputs is made.

Note 3.6. The expected number of trials of $\minpol$ used in the above process is bounded by a constant, and the expected running time of each trial is as described in Theorem 3.4. Of course, since expectations are not generally multiplicative, this is not quite enough to conclude that the expected cost of the entire process is as stated above. However, this is the case, since the Las Vegas algorithms used here are performing Bernoulli trials.

In particular, tracing back through the code shown in Figure 1, and reviewing the description of function $\minpolvec$ in Section 2.2 following Fact 2.8, one can confirm that the desired complexity result will follow if it can be established that the expected number of applications of the function $\minpolvec$ is bounded by a constant. Now, since $\minpol$ requires $2c$ applications of $\minpolvec$ for the constant $c = \trialbound(k) \leq 6$, the probability that $2c(i - 1)^2 - 1$ or more applications of $\minpolvec$ are needed is at most

$$\frac{2c(i-1)^2+1}{2^p},$$

because this condition would imply either that one of the first $2c(i - 1)$ applications of $\minpolvec$ requires $i$ or more applications of $\minpolvec$, or that the entire process requires $i$ or more applications of $\minpol$. Over-approximating the probability that between $2c(i - 1)^2 - 1$ and
$2ci^2$ applications of minpolvec-1/2 are needed by the probability that at least $2c(i-1)^2-1$ are, one can bound the expected number of applications of minpolvec-1/2 by

$$\sum_{i \geq 1} \frac{2c(i-1)+1}{2^i} \in O(1),$$

as required.

### 3.2 Application to Computation of the Frobenius Form

In this section, it is established that the machinery developed so far can be used to find the Frobenius form of a matrix. These results depend on the existence and uniqueness of the Frobenius form of a matrix as established, for example, by Gantmacher [9].

**Lemma 3.7.** Let $A \in F^{n \times n}$, let $X \in F^{n \times n}$ be nonsingular, and suppose

$$X^{-1}AX = \begin{bmatrix} C_{f_1} & & 0 \\ & C_{f_2} & \\ & & \ddots \\ 0 & & & C_{f_k} \\ & & & B \end{bmatrix}$$

where $C_{f_1}, C_{f_2}, \ldots, C_{f_k}$ are companion matrices of polynomials $f_1, f_2, \ldots, f_k$ of positive degree such that $f_{i+1}$ divides $f_i$ for $1 \leq i \leq k-1$, and such that $B \in F^{m \times m}$ for some integer $m \leq n$.

If $f_k(B) = 0$ then the invariant factors of $A$ are $f_1, f_2, \ldots, f_k, g_1, g_2, \ldots, g_l$ where $g_1, g_2, \ldots, g_l$ are the invariant factors of $B$ and, if $Y \in F^{m \times m}$ is a Frobenius transition matrix for $B$, then

$$Z = X \begin{bmatrix} I_{n-m} \\ Y \end{bmatrix}$$

is a Frobenius transition matrix for $A$.

Conversely, if $f_1, f_2, \ldots, f_k$ are the first $k$ invariant factors of $A$ then $f_k(B) = 0$.

**Proof.** Suppose $B$ has invariant factors $g_1, g_2, \ldots, g_l$ and a Frobenius transition matrix $Y \in F^{m \times m}$, so that $g_{j+1}$ divides $g_j$ for $1 \leq j \leq l-1$ and

$$Y^{-1}BY = \begin{bmatrix} C_{g_1} & & 0 \\ & C_{g_2} & \\ & & \ddots \\ 0 & & & C_{g_l} \end{bmatrix}.$$
Then it is easily verified that

\[
Z^{-1}AZ = \left[ I_{n-m} \ Y \right]^{-1} X^{-1}AX \left[ I_{n-m} \ Y \right]
\]

\[
= \left[ I_{n-m} \ Y \right]^{-1} \begin{bmatrix}
C_{f_1} & \cdots & 0 \\
0 & C_{f_k} & B \\
0 & C_{f_k} & Y^{-1}BY
\end{bmatrix} \left[ I_{n-m} \ Y \right]
\]

\[
= \begin{bmatrix}
C_{f_1} & 0 \\
\cdots & C_{f_k} & \cdots \\
0 & \cdots & C_{g_l}
\end{bmatrix}
= C,
\]

where

\[
C = \begin{bmatrix}
C_{f_1} & \cdots & \cdots & C_{f_k} \\
\cdots & \cdots & \cdots & \cdots \\
0 & \cdots & \cdots & C_{g_l}
\end{bmatrix}.
\]

Now, if \(g_1\) divides \(f_k\) then the above matrix \(C\) is in Frobenius form and it follows by the uniqueness of this matrix form that \(f_1, f_2, \ldots, f_k, g_1, g_2, \ldots, g_l\) are the invariant factors of \(A\) and that \(Z\) is a transition matrix for \(A\), as claimed.

Conversely, suppose \(f_1, f_2, \ldots, f_k\) are the first \(k\) invariant factors of \(A\) but that \(f_k(B) \neq 0\), and let \(g\) be the minimum polynomial of \(B\). Let \(i\) be the largest integer such that \(g\) divides \(f_i\), choosing \(i = 0\) if \(g\) does not divide \(f_1\), so that \(0 \leq i \leq k - 1\). Let

\[
\hat{B} = \begin{bmatrix}
C_{f_{i+1}} & C_{f_{i+1}} & \cdots & 0 \\
C_{f_{i+1}} & \cdots & C_{f_k} & B \\
0 & \cdots & \cdots & \cdots
\end{bmatrix},
\]

so that

\[
X^{-1}AX = \begin{bmatrix}
C_{f_1} & C_{f_2} & \cdots & 0 \\
\cdots & \cdots & \cdots & \cdots \\
0 & \cdots & \cdots & C_{f_i} \\
& & & \hat{B}
\end{bmatrix}.
\]

(13)

Set \(h = \text{lcm}(g, f_{i+1})\); then \(h\) divides \(f_i\) if \(i > 0\), since \(g\) divides \(f_i\) by the choice of \(i\) and \(f_{i+1}\) divides \(f_i\) since \(f_i\) and \(f_{i+1}\) are successive invariant factors of \(A\). On the other hand, \(h \neq f_{i+1}\) since \(g\) does not divide \(f_{i+1}\).

Clearly, \(h(\hat{B}) = 0\) since \(h(C_{f_j}) = 0\) for \(i + 1 \leq j \leq k\) and since \(h(B) = 0\) as well. On the other hand, if \(h\) is any proper divisor of \(h\) then \(h(\hat{B}) \neq 0\), since either \(h\) does not divide \(f_{i+1}\) or \(h\) does not
In order to prove the first part of the claim it is necessary and sufficient to show that the matrices for a matrix and let

Thus $h$ is the minimum polynomial of $\hat{B}$.

However, since equation (13) above is satisfied, and $f_i(\hat{B}) = 0$, the first part of the claim can be applied to conclude that the $i + 1$st invariant factor of $A$ is the minimum polynomial $h$ of $B$, contradicting the fact that $h \neq f_{i+1}$. Therefore $f_k(B) = 0$ as claimed. □

**Lemma 3.8.** Let $A \in \mathbb{F}^{n \times n}$ and let $f_1, f_2, \ldots, f_k \in \mathbb{F}[x]$ be polynomials with positive degree such that $f_i$ is divisible by $f_{i+1}$ for $1 \leq i \leq k - 1$. Let $d_i$ be the degree of $f_i$ for all $i$.

Let $u_1, u_2, \ldots, u_k, v_1, v_2, \ldots, v_k \in \mathbb{F}^{n \times 1}$ be vectors such that

$$\text{minpol}(A^T, u_i) = \text{minpol}(u_i^T, A, v_i) = \text{minpol}(A, v_i) = f_i \quad \text{for } 1 \leq i \leq k$$

and

$$u_i^T A^l v_j = 0 \quad \text{for all } l \geq 0 \text{ whenever } 1 \leq i, j \leq k \text{ and } i \neq j.$$ (15)

Then the vectors

$$v_1, Av_1, \ldots, A^{d_i-1}v_1, v_2, Av_2, \ldots, A^{d_i-1}v_2, \ldots, v_k, Av_k, \ldots, A^{d_k-1}v_k$$

are linearly independent, as are the vectors

$$u_1, A^T u_1, \ldots, (A^T)^{d_i-1}u_1, u_2, A^T u_2, \ldots, (A^T)^{d_i-1}u_2, \ldots, u_k, A^T u_k, \ldots, (A^T)^{d_k-1}u_k.$$ (14)

Furthermore, if $m = n - \sum_{i=1}^{k} d_i > 0$ then there exist vectors $\mu_1, \mu_2, \ldots, \mu_m, \nu_1, \nu_2, \ldots, \nu_m \in \mathbb{F}^{n \times 1}$ such that

$$u_i^T A^l v_j = \mu_j A^l v_i = 0$$ (16)

for $1 \leq i \leq k$, $1 \leq j \leq m$, and all $l \geq 0$, such that

$$u_1, A^T u_1, \ldots, (A^T)^{d_i-1}u_1, \ldots, u_k, A^T u_k, \ldots, (A^T)^{d_k-1}u_k, \mu_1, \mu_2, \ldots, \mu_m$$

and

$$v_1, Av_1, \ldots, A^{d_i-1}v_1, \ldots, v_k, Av_k, \ldots, A^{d_k-1}v_k, \nu_1, \nu_2, \ldots, \nu_m$$

are both bases for $\mathbb{F}^{n \times 1}$. Every vector $\mu \in \mathbb{F}^{n \times 1}$ such that $\mu^T A^l v_i = 0$ for all $l \geq 0$ and $1 \leq i \leq k$ is an $\mathbb{F}$-linear combination of $\mu_1, \mu_2, \ldots, \mu_m$, and every vector $\nu \in \mathbb{F}^{n \times 1}$ such that $u_i^T A^l \nu = 0$ for all $l \geq 0$ and $1 \leq i \leq k$ is an $\mathbb{F}$-linear combination of $\nu_1, \nu_2, \ldots, \nu_m$.

**Proof.** Let

$$X = [u_1 \ A^T u_1 \ldots \ (A^T)^{d_i-1}u_1 \ldots \ u_k \ (A^T)u_k \ldots \ (A^T)^{d_k-1}u_k] \in \mathbb{F}^{n \times (n-m)}$$

and let

$$Y = [v_1 \ Av_1 \ldots \ A^{d_i-1}v_1 \ldots \ v_k \ Av_k \ldots \ A^{d_k-1}v_k] \in \mathbb{F}^{n \times (n-m)}.$$ (14)

In order to prove the first part of the claim it is necessary and sufficient to show that the matrices $X$ and $Y$ both have full rank $n - m$. Now, the orthogonality condition (15) implies that $X^T Y = H$ for a matrix

$$H = \begin{bmatrix} H_1 & 0 & \cdots & 0 \\ H_2 & & & \\ \vdots & & & \\ 0 & & & H_k \end{bmatrix} \in \mathbb{F}^{(n-m) \times (n-m)},$$

28
Then, if $1 \leq i \leq k$ and that for $1 \leq i \leq l \leq m$ where $H_i$ is a Hankel matrix

$$H_i = \begin{bmatrix}
    u_i^T v_i & u_i^T A v_i & \ldots & u_i^T A^{d_i-1} v_i \\
    u_i^T A v_i & u_i^T A^2 v_i & \ldots & u_i^T A^{d_i-1} v_i \\
    \vdots & \vdots & \ddots & \vdots \\
    u_i^T A^{d_i-1} v_i & u_i^T A^{d_i} v_i & \ldots & u_i^T A^{2d_i-2} v_i \\
\end{bmatrix} \in F^{d_i \times d_i} \quad (17)$$

for $1 \leq i \leq k$. Since $\minpol(u_i^T, A, v_i)$ has degree $i$ by condition (14), above, it follows from Lemma 1 of Kaltofen and Pan [14] that $H_i$ is nonsingular, for all $i$. Thus $H$ is nonsingular as well and $X$ and $Y$ must have full rank $n - m$, as needed.

Since $X$ has rank $n - m$, its left kernel has dimension $m$; let $\nu_1, \nu_2, \ldots, \nu_m \in F^{n \times 1}$ be a basis for the set of vectors $\{x \in F^{n \times 1} : x^T X = 0\}$. These vectors are clearly linearly independent by construction. It is clear by the construction of $u_i$ that orthogonality relations that involve them in equation (16), above.

Finally, suppose $c_{i,j}, e_l \in F$ for $1 \leq i \leq k$, $0 \leq j \leq d_i - 1$, and $1 \leq l \leq m$ such that

$$\sum_{i=1}^{k} \sum_{j=0}^{d_i-1} c_{i,j} (A^j v_i) + \sum_{l=1}^{m} e_l \nu_l = 0. \quad (19)$$

Then, if $1 \leq i \leq k$, then it follows by the choice of $u_1, u_2, \ldots, u_k$ and $\nu_1, \nu_2, \ldots, \nu_m$ that if $1 \leq h \leq k$, $h \neq i$, and $r \geq 0$, then

$$u_i^T A^r \left( \sum_{j=0}^{d_h-1} c_{h,j} (A^j v_h) \right) = \sum_{j=0}^{d_h-1} c_{h,j} (u_i^T A^{j+r} v_h) = \sum_{j=0}^{d_h-1} c_{h,j} \cdot 0 = 0,$$

and that

$$u_i^T A^r e_l \nu_l = e_l u_i^T A^r \nu_l = e_l \cdot 0 = 0$$

for $1 \leq l \leq m$ as well. Since

$$u_i^T A^r \left( \sum_{h=1}^{k} \sum_{j=0}^{d_h-1} c_{h,j} (A^j v_h) + \sum_{l=1}^{m} e_l \nu_l \right) = 0$$

by equation (19), above, it follows that

$$u_i^T A^r \left( \sum_{j=0}^{d_i-1} c_{i,j} (A^j v_i) \right) = \sum_{j=0}^{d_i-1} c_{i,j} (u_i^T A^{j+r} v_i) = 0$$
too, for all $r$. Now, since the minimum polynomial of the sequence
\[ u_i^T v_i, u_i^T A v_i, u_i^T A^2 v_i \ldots \]
has degree $d_i$, this implies that
\[ c_{i,0} = c_{i,1} = \cdots = c_{i,d_i-1} = 0. \]
Since this holds for all $i$, it now follows by equation (19) that
\[ \sum_{i=1}^{m} e_i v_i = 0 \]
as well, so that $e_1 = e_2 = \cdots = e_m = 0$, by the linear independence of $v_1, v_2, \ldots, v_m$. Therefore equation (19) is only satisfied when $c_{i,j} = e_l = 0$ for all $i$, $j$ and $l$, so that
\[ v_1, A v_1, \ldots, A^{d_1-1} v_1, v_k, A v_k, \ldots, A^{d_k-1} v_k, v_1, v_2, \ldots, v_m \]
are linearly independent as required. At this point, all the properties claimed for the vectors $v_1, v_2, \ldots, v_m$ have been proved.

The same argument, applied to $Y$ instead of $X$, establishes the existence of vectors $\mu_1, \mu_2, \ldots, \mu_m$ with the needed properties as well, and completes the proof.

Suppose next that the polynomials $f_1, f_2, \ldots, f_k$ and the vectors $u_1, u_2, \ldots, u_k, v_1, v_2, \ldots, v_k, \mu_1, \mu_2, \ldots, \mu_m$ and $v_1, v_2, \ldots, v_m$ satisfy the conditions given in the previous lemma. Set $\hat{X}, \hat{Y} \in \mathbb{F}^{n \times n}$ to be matrices with columns
\[ u_1, A^T u_1, \ldots, (A^T)^{d_1-1} u_1, \ldots, v_k, A v_k, \ldots, (A^T)^{d_k-1} v_k, \mu_1, \mu_2, \ldots, \mu_m \]
and
\[ v_1, A v_1, \ldots, A^{d_1-1} v_1, \ldots, v_k, A v_k, \ldots, A^{d_k-1} v_k, v_1, v_2, \ldots, v_m \]
respectively. Then $\hat{X}$ and $\hat{Y}$ are both nonsingular, since the columns of each form a basis for $\mathbb{F}^{n \times 1}$, and the orthogonality relations included in the lemma imply that
\[
\hat{X}^T \hat{Y} = \begin{bmatrix}
H_1 & 0 & \cdots & 0 \\
H_2 & \ddots & \ddots & \\
0 & \ddots & \ddots & \ddots \\
0 & \cdots & \cdots & H_k \\
\end{bmatrix}
\]
where $H_i \in \mathbb{F}^{d_i \times d_i}$ is the nonsingular Hankel matrix shown in equation (17), above, and where
\[ C_k = [\mu_1 \mu_2 \cdots \mu_m]^T [v_1 \nu_2 \cdots \nu_m] \in \mathbb{F}^{m \times m}. \]
Since $\hat{X}$ and $\hat{Y}$ are nonsingular, $\hat{X}^T \hat{Y}$ and $C_k$ must clearly be nonsingular as well. The orthogonality relations also imply that
\[
\hat{X}^T A Y = \begin{bmatrix}
A_1 & 0 & \cdots & 0 \\
A_2 & \ddots & \ddots & \\
0 & \ddots & \ddots & \ddots \\
0 & \cdots & \cdots & A_k \\
\end{bmatrix},
\]
for matrices

\[ A_i = [u_i \ A^T \ u_i \ \ldots \ (A^T)^{d_i-1} \ u_i]^T \cdot A \cdot [v_i \ A \ v_i \ \ldots \ A^{d_i-1} \ v_i] \in F^{d_i \times d_i} \]

for \( 1 \leq i \leq k \), and for

\[ \hat{A}_k = [\mu_1 \ \mu_2 \ \ldots \ \mu_m]^T \cdot A \cdot [\nu_1 \ \nu_2 \ \ldots \ \nu_m] \in F^{m \times m} \].

Now, equation (20) implies that

\[
\begin{bmatrix}
H_1^{-1} & 0 \\
H_2^{-1} & \cdots \\
\vdots & \\
0 & H_k^{-1} \\
C^{-1} & \\
\end{bmatrix}
\cdot \hat{X}^T = \hat{Y}^{-1},
\]

so that

\[
\hat{Y}^{-1} A \hat{Y} = 
\begin{bmatrix}
H_1^{-1} & 0 \\
H_2^{-1} & \cdots \\
\vdots & \\
0 & H_k^{-1} \\
C^{-1} & \\
\end{bmatrix}
\begin{bmatrix}
\hat{X}^T A \hat{Y} \\
\end{bmatrix}
\]

\[
= 
\begin{bmatrix}
H_1^{-1} A_1 & 0 \\
H_2^{-1} A_2 & \cdots \\
\vdots & \\
0 & H_k^{-1} A_k \\
C^{-1} & \\
\end{bmatrix}
= 
\begin{bmatrix}
C_{f_1} & 0 \\
C_{f_2} & \cdots \\
\vdots & \\
0 & C_{f_k} \\
\end{bmatrix}
\begin{bmatrix}
C^{-1} A_k \\
\end{bmatrix}
\]

noting that \( H_i^{-1} A_i = C_{f_i} \) for \( 1 \leq i \leq k \) by inspection of the first \( n - m \) columns of \( \hat{Y} \) and using the fact that \( \minpol(A, v_i) = f_i \) for all \( i \).

In this case, if \( f_1, f_2, \ldots, f_k \) are the first \( k \) invariant factors of \( A \) then the second half of Lemma 3.7 implies that \( f_k(C_k^{-1} \hat{A}_k) = 0 \), so that the first half of the lemma implies that the remaining invariant factors of \( A \) are the invariant factors of \( C_k^{-1} \hat{A}_k \). In particular, the minimum polynomial of \( C_k^{-1} \hat{A}_k \) is the \( k + 1 \)st invariant factor \( f_{k+1} \) of \( A \).

Equation (20) and the above expression for \( \hat{Y}^{-1} A \hat{Y} \) can be used to establish that

\[
\hat{X}^{-1} A^T \hat{X} = 
\begin{bmatrix}
(H_1 C_{f_1} H_1^{-1})^T \\
(H_2 C_{f_2} H_2^{-1})^T \\
\vdots \\
(H_k C_{f_k} H_k^{-1})^T \\
(\hat{A}_k C_k^{-1})^T \\
\end{bmatrix}
\]

31
and, since $A^T$ has the same invariant factors as $A$, that the minimum polynomial of $(\hat{A}_kC_k^{-1})^T$ is the $k + 1$\textsuperscript{st} invariant factor of $A$ as well.

**Theorem 3.9.** Let $A \in \mathbb{F}^{n \times n}$, let $k \geq 0$, and suppose $A$ has at least $k$ nontrivial invariant factors. Let $f_1, f_2, \ldots, f_k \in \mathbb{F}[x]$ be the first $k$ invariant factors of $A$. Let $u_1, u_2, \ldots, u_k, v_1, v_2, \ldots, v_k \in \mathbb{F}^{n \times 1}$ be vectors such that

$$
\text{minpol}(A, v_k) = \text{minpol}(A, v_{k-1}) = \cdots = \text{minpol}(A, v_1)
$$

for $1 \leq i \leq k$, and

$$
u_i^T A^j v_j = 0
$$

for every integer $l \geq 0$ whenever $1 \leq i, j \leq k$ and $i \neq j$. Let

$$
U_{k+1} = \{u \in \mathbb{F}^{n \times 1} \mid u_i^T A^j v_j = 0 \text{ for } l \geq 0 \text{ and } 1 \leq j \leq k\}
$$

and let

$$
V_{k+1} = \{v \in \mathbb{F}^{n \times 1} \mid u_i^T A^j v = 0 \text{ for } l \geq 0 \text{ and } 1 \leq j \leq k\}.
$$

Then

(a) If $A$ has exactly $k$ invariant factors then $U_{k+1} = V_{k+1} = (0)$.

(b) Otherwise, $U_{k+1}$ and $V_{k+1}$ are $A$-complementary subspaces such that

$$
\text{minpol}(A^T, U_{k+1}) = \text{minpol}(U_{k+1}^T, A, V_{k+1}) = \text{minpol}(A, V_{k+1})
$$

is the $k + 1$\textsuperscript{st} invariant factor of $A$.

**Proof.** If $A$ has at most $k$ nontrivial invariant factors then the union of bases for the spaces $K_{u_1, v_1}^{(L)}, K_{u_2, v_2}^{(L)}, \ldots, K_{u_k, v_k}^{(L)}$ form a basis for $A$ and, since an arbitrary element $v$ of $V_{k+1}$ is orthogonal to each of the elements of this basis, $v = 0$ and $V_{k+1} = (0)$. A similar argument (in which $K_{u_1, v_1}^{(R)}, K_{u_2, v_2}^{(R)}, \ldots, K_{u_k, v_k}^{(R)}$ are considered, instead) establishes that $U_{k+1} = (0)$ in this case as well.

Otherwise, it is clear by their definitions that $U_{k+1}$ is $A^T$-invariant and $V_{k+1}$ is $A$-invariant. Let $d_i$ be the degree of $f_i$ for $1 \leq i \leq k$, let $m = n - \sum_{i=1}^{k} d_i$, and note that the last $m$ columns of the matrix $\hat{Y}$, discussed earlier in this section, form a basis for $V_{k+1}$. The above argument (in particular, the expression given above for $\hat{Y}^{-1} A \hat{X}$ and the derivation of the minimum polynomial of $C_k^{-1} \hat{A}_k$) implies that

$$
\text{minpol}(A, V_{k+1}) = f_{k+1}
$$

is the $k + 1$\textsuperscript{st} invariant factor of $A$, for it establishes both that $\text{minpol}(A, v) = f_k$ is divisible by this polynomial for every element $v$ of $V_{k+1}$ and that $\text{minpol}(A, v_k) = f_k$ is equal to this polynomial for at least one element $v_k$ of $V_{k+1}$.

Since $A^T$ has the same invariant factors as $A$, a similar argument (using the above expression for $\hat{X}^{-1} A^T \hat{X}$) establishes that $\text{minpol}(A^T, U_{k+1}) = f_{k+1}$ as well.

Finally, let $v_{k+1} \in V_{k+1}$ be as above, and let $u \in \mathbb{F}^{n \times 1}$ be a vector such that

$$
\text{minpol}(u^T, A, v_{k+1}) = \text{minpol}(A, v_{k+1}) = f_{k+1}.
$$

Lemma 3.8 implies that the union of bases for $K_{u_1, v_1}^{(L)}, K_{u_2, v_2}^{(L)}, \ldots, K_{u_k, v_k}^{(L)}$ and for $U_{k+1}$ forms a basis for $\mathbb{F}^{n \times 1}$, so that there exist vectors $w_1 \in K_{u_1, v_1}^{(L)}, w_2 \in K_{u_2, v_2}^{(L)}, \ldots, w_k \in K_{u_k, v_k}^{(L)}$ and $u_{k+1} \in U_{k+1}$ such that $u = w_1 + w_2 + \cdots + w_k + u_{k+1}$. Since the definition of $V_{k+1}$ implies that $w_i^T A^j v_{k+1} = 0$
for every integer \( l \geq 0 \) and for \( 1 \leq i \leq k \), it follows that \( u^T A^l v_{k+1} = u_{k+1}^T A^l v_{k+1} \) for every integer \( l \geq 0 \), so that
\[
\minpol(u_{k+1}^T, A, v_{k+1}) = \minpol(u^T, A, v_{k+1}) = f_{k+1}.
\]
Now, since the polynomial \( \minpol(U_{k+1}^T, A, V_{k+1}) \) is both divisible by \( \minpol(u_{k+1}^T, A, v_{k+1}) \) and a divisor of \( \minpol(A, V_{k+1}) \), it follows that
\[
\minpol(U_{k+1}^T, A, V_{k+1}) = f_{k+1}
\]
as well, as required to establish that \( U_{k+1} \) and \( V_{k+1} \) are \( A \)-complementary and to complete the proof. \( \square \)

4 Algorithms for the Frobenius Form

Three versions of an algorithm for the Frobenius form of a matrix that use the techniques from the previous sections will be introduced: Algorithms for computations over small fields and large fields that can be used to establish interesting results when implemented with standard arithmetic, and, finally, a version improving the known bound on the asymptotic complexity of the problem in the small field case.

All versions of the algorithm will receive or manipulate the following information.

- \( A \in F^{n \times n} \) is, of course, in the input matrix.
- \( k \) is a nonnegative integer less than or equal to the number of nontrivial invariant factors of \( A \).
- \( f_i \in F[x] \) is a monic polynomial, for \( 1 \leq i \leq k \), such that \( f_{i+1} \) divides \( f_i \) for \( 1 \leq i \leq k - 1 \).
- \( d_i = \text{deg}(f_i) \) for \( 1 \leq i \leq k \).
- \( d = d_1 + d_2 + \cdots + d_k \).
- \( u_i, v_i \in F^{n \times 1} \) are vectors such that
\[
\minpol(A^T, u_i) = \minpol(u_i^T, A, v_i) = \minpol(A, v_i) = f_i
\]
for \( 1 \leq i \leq k \) and, furthermore, such that
\[
u_i^T A^l v_j = 0
\]
for all \( l \geq 0 \) and all integers \( i, j \) such that \( 1 \leq i, j \leq k \) and \( i \neq j \).
- \( u_{i,1}, u_{i,2}, \ldots, u_{i,d_i} \in K_{u_i,v_i}^{(L)} \) and \( v_{i,1}, v_{i,2}, \ldots, v_{i,d_i} \in K_{u_i,v_i}^{(R)} \) are dual bases for \( A, u_i \) and \( v_i \), for \( 1 \leq i \leq k \) (see Section 2.1 and, in particular, Definition 2.3 for a discussion of “dual bases”).
- The values \( s_{i,j} = u_{i,j}^T v_{i,j} \) for \( 1 \leq j \leq d_i \) and \( 1 \leq i \leq k \). Note that Definition 2.3 and the above descriptions of \( u_{i,j} \) and \( v_{i,j} \) imply that \( s_{i,j} \neq 0 \).

It will also be necessary for these algorithms to select values from the vector spaces
\[
U_{(k+1)} = \{ u \in F^{n \times 1} \mid u^T A^l v_j = 0 \text{ for } l \geq 0 \text{ and } 1 \leq j \leq k \}
\]
function randU-small()
begin function
Choose a vector \( \hat{u} \) uniformly and randomly from \( F^{n \times 1} \)
return \( \hat{u} - \sum_{i=1}^{k} \sum_{j=1}^{d_i} \frac{u^T_i v_{i,j} s_{i,j}}{s_{i,j}} u_{i,j} \)
end function

function randV-small()
begin function
Choose a vector \( \hat{v} \) uniformly and randomly from \( F^{n \times 1} \)
return \( \hat{v} - \sum_{i=1}^{k} \sum_{j=1}^{d_i} \frac{u^T_i v_{i,j} s_{i,j}}{s_{i,j}} v_{i,j} \)
end function

Figure 5: Functions \texttt{randU-small} and \texttt{randV-small}

\[
V_{(k+1)} = \{ v \in F^{n \times 1} \mid u^T_i A^l v = 0 \text{ for } l \geq 0 \text{ and } 1 \leq j \leq k \}.
\]

It follows by the definition of \( v_{i,j} \) (respectively, \( u_{i,j} \)) that \( U_{(k+1)} \) is the set of vectors that are orthogonal to \( v_{i,j} \) for \( 1 \leq i \leq k \) and \( 1 \leq j \leq d_i \), and that \( V_{(k+1)} \) is the set of vectors that are orthogonal to \( u_{i,j} \) for \( 1 \leq i \leq k \) and \( 1 \leq j \leq d_i \).

Each algorithm will repeatedly apply the function \texttt{minpolspace} from Section 3.1 with input vectors chosen from \( U_{(k+1)} \) and \( V_{(k+1)} \) to either increment \( k \) and extend the above sequences of values or, in the first and last versions of the algorithm, improve the values that have been generated already.

Each algorithm will terminate when \( d = n \). At this point, it will follow by a straightforward argument that \( f_1, f_2, \ldots, f_k \) are the invariant factors of \( A \). It is clear that the matrix \( V \in F^{n \times n} \) with columns \( v_1, Av_1, \ldots, A^{d_1-1}v_1, \ldots, v_k, Av_k, \ldots, A^{d_k-1}v_k \) is a Frobenius transition matrix for \( A \). Thus a Frobenius transition matrix for \( A \) can also be computed, after applying any of the algorithms described here, using at most \( n-1 \) additional matrix-vector products by the matrix \( A \).

4.1 An Algorithm for Computations over Small Fields

Suppose \( F \) is a finite field; then the functions \texttt{randU-small} and \texttt{randV-small}, shown in Figure 5, will be used to uniformly select elements of the above spaces \( U_{(k+1)} \) and \( V_{(k+1)} \).

Lemma 4.1. Functions \texttt{randU-small} and \texttt{randV-small} generate uniformly and randomly selected elements of the subspaces \( U_{(k+1)} \) and \( V_{(k+1)} \) respectively. Each function chooses \( O(n) \) elements uniformly and independently from \( F \) and performs \( O(n^2) \) additional operations over \( F \).

Proof. Since inner products are linear operators, it is clear by inspection of the code that function \texttt{randU-small} generates a uniformly and randomly selected \( F \)-linear combination of the elements of a
spanning set for $U_{(k+1)}$, namely, the elements

$$e_h - \sum_{i=1}^{k} \sum_{j=1}^{d_i} \frac{d_i}{s_i,j} u_i,j,$$

where $e_h$ is the $h^{th}$ unit vector (whose $h^{th}$ entry is one and whose other entries are zero), for $1 \leq h \leq n$. Thus the function returns a uniformly and randomly selected element of $U_{(k+1)}$, as claimed. The correctness of `randV-small` follows by the same argument.

The claimed complexity bounds follow by a inspection of the code, using the fact that $d = \sum_{i=1}^{k} d_i \leq n$. \hfill $\Box$

The function `frobenius-small` shown in Figure 6 can be used to find the Frobenius form of a given matrix $A \in \mathbb{F}^{n \times n}$ with entries in a finite field $\mathbb{F}$. The algorithm maintains the data described at the beginning of Section 4. While $d < n$, sequences of vectors $\hat{u}_1, \hat{u}_2, \ldots, \hat{u}_l$ and $\hat{v}_1, \hat{v}_2, \ldots, \hat{v}_l$ are accumulated, to serve as inputs for `minpolspace` (in lines 3–12), and an upper bound “$\text{bnd}$” on the degree of

$$\text{lcm} \left( \text{lcm} \minpol(A^T, \hat{u}_1), \text{lcm} \minpol(A, \hat{u}_i) \right)$$

is obtained — namely, the degree of the last polynomial $f_j$ found (and maintained) by the algorithm such that $f_j(A^T) \hat{u}_i = f_j(A) \hat{v}_i = 0$ for $1 \leq i \leq l$, or $n$ if no such polynomial exists. The variable “dropped” is used to keep track of whether any of the guessed invariant factors of $A$ have been discarded, and is used to decide whether the output of `minpolspace` should be used to improve an existing guess or add a new polynomial to the sequence (in lines 13–18).

**Theorem 4.2.** Suppose $\mathbb{F}$ is a finite field and $A \in \mathbb{F}^{n \times n}$.

Function `frobenius-small` terminates with probability one, on input $A$, and returns the number $k$ of invariant factors of $A$, the invariant factors $f_1, f_2, \ldots, f_k$, and vectors $u_1, u_2, \ldots, u_k$ and $v_1, v_2, \ldots, v_k$ such that $f_i v_j = 0$ for $1 \leq i \leq k$ and $i \neq j$.

The expected number of matrix-vector products (using the matrices $A$ or $A^T$) used by this algorithm is in $O(n^3)$, and the expected number of additional operations over $\mathbb{F}$ used by the algorithm is in $O(kn^2) \leq O(n^3)$, if $A$ has $k$ nontrivial invariant factors.

**Proof.** The algorithm begins with an empty sequence of guessed invariant factors and associated vectors. An inspection of the code, and the functions shown in Figures 1–4, confirms that at the end of each execution of the outer loop, a set of monic polynomials $f_1, f_2, \ldots, f_k$ and vectors $u_1, u_2, \ldots, u_k$ and $v_1, v_2, \ldots, v_k$ have been generated such that $f_i v_j = 0$ for $1 \leq i \leq k$, $i \neq j$, and, furthermore, such that $u_i^T A v_j = 0$ for $1 \leq i, j \leq k$, $i \neq j$, and for all $l \geq 0$ — for the function `minpolspace` is guaranteed to return a polynomial that is divisible by the least common multiple of the minimum polynomials of its input vectors, which ensures that the divisibility relationship for the polynomials is maintained, and the input vectors are selected from $A$-invariant (and $A^T$-invariant) subspaces that are orthogonal to the vectors that have been generated so far. If the algorithm terminates then it does so when the sum
function \text{frobenius-small}(A)

\begin{align*}
\text{begin function} \\
1. & k := 0; d := 0 \\
2. & \text{while } d < n \text{ do} \\
3. & \quad \text{if } k > 0 \text{ then } \text{bnd} := d_k \text{ else } \text{bnd} := n \text{ end if} \\
4. & \quad l := 0; \text{dropped} := \text{false} \\
5. & \text{while } l < \text{trialbound(\text{bnd})} \text{ do} \\
6. & \quad l := l + 1 \\
7. & \quad \hat{u}_l := \text{randU-small}(); \hat{v}_l := \text{randV-small}() \\
8. & \quad \text{while } k > 0 \text{ and } (f_k(A^T)\hat{u}_l \neq 0 \text{ or } f_k(A)\hat{v}_l \neq 0) \text{ do} \\
9. & \quad \quad \text{dropped} := \text{true}; d := d - d_k; k := k - 1 \\
10. & \quad \quad \text{if } k > 0 \text{ then } \text{bnd} := d_k \text{ else } \text{bnd} := n \text{ end if} \\
11. & \text{end while} \\
12. & \text{end while} \\
13. & \text{if } \text{dropped} \text{ then} \\
14. & \quad u_{k+1}, v_{k+1}, f_{k+1} := \text{mergev}(A, u_{k+1}, v_{k+1}, f_{k+1}, \text{minpolspace}(A; \text{bnd}; \hat{u}_1, \hat{u}_2, \ldots, \hat{u}_l; \hat{v}_1, \hat{v}_2, \ldots, \hat{v}_l)) \\
15. & \text{else} \\
16. & \quad u_{k+1}, v_{k+1}, f_{k+1} := \text{minpolspace}(A; \text{bnd}; \hat{u}_1, \hat{u}_2, \ldots, \hat{u}_l; \hat{v}_1, \hat{v}_2, \ldots, \hat{v}_l) \\
17. & \text{end if} \\
18. & d_{k+1} := \text{deg}(f_{k+1}); d := d + d_{k+1} \\
19. & u_{k+1,1}, u_{k+1,2}, \ldots, u_{k+1,d_{k+1}}; v_{k+1,1}, v_{k+1,2}, \ldots, v_{k+1,d_{k+1}} := \text{dualbasis}(u_{k+1}^T, A, v_{k+1}, d_{k+1}) \\
20. & \text{for } i := 1 \ldots d_{k+1} \text{ do } s_{k+1,i} := u_{k+1,i}^T v_{k+1,i+1} \text{ end for} \\
21. & k := k + 1 \\
22. & \text{end while} \\
23. & \text{return } k; f_1, f_2, \ldots, f_k; u_1, u_2, \ldots, u_k; v_1, v_2, \ldots, v_k \\
\text{end function}
\end{align*}

Figure 6: Function \text{frobenius-small}

of the degrees of the polynomials in the sequence equals \( n \), at which point it follows by Lemma 3.8 that (for \( d_i = \text{deg}(f_i) \)) the vectors

\[ v_1, Av_1, \ldots, A^{d_1-1}, v_2, Av_2, \ldots, A^{d_2-1}, \ldots, v_k, Av_k, \ldots, A^{d_k-1} \]

are linearly independent and (since there are \( n \) of them), form a basis for \( \mathbb{F}^{n \times 1} \). Now, if \( Y \in \mathbb{F}^{n \times n} \) is the nonsingular matrix with these entries as columns then clearly \( Y^{-1}AY \) is a matrix in Frobenius form with invariant factors \( f_1, f_2, \ldots, f_k \), so \( A \) has these invariant factors as well.

In order to prove the claims about the complexity of the algorithm it helps to think of the algorithm as proceeding in \( k \) stages, where the \( i^{th} \) stage (for \( 1 \leq i \leq k \)) is used to discover the \( i^{th} \)
function randU-large()
begin function
    Choose the entries of a vector \( \hat{u} \in \mathbb{F}^{n \times 1} \) uniformly and independently from \( S \)
    return \( \hat{u} - \sum_{i=1}^{k} \sum_{j=1}^{d_i} \frac{\hat{u}^T v_{i,j} \hat{u}}{s_{i,j}} u_{i,j} \)
end function

function randV-large()
begin function
    Choose the entries of a vector \( \hat{v} \in \mathbb{F}^{n \times 1} \) uniformly and independently from \( S \)
    return \( \hat{v} - \sum_{i=1}^{k} \sum_{j=1}^{d_i} \frac{\hat{v}^T u_{i,j} \hat{v}}{s_{i,j}} v_{i,j} \)
end function

Figure 7: Functions randU-large and randV-large

The proof that these functions return values as described in the next claim is, essentially, identical to that of Lemma 4.1; the spanning set mentioned in the lemma is the same as the one used in that lemma. A proof of the complexity bound stated here is completely straightforward.

**Lemma 4.3.** Functions randU-large and randV-large return uniformly and randomly selected \( S \)-linear combinations of spanning sets for the subspaces \( U_{k+1} \) and \( V_{k+1} \), respectively. Each function chooses \( n \) elements uniformly and independently from \( S \), and performs at most \( 4n^2 \) additional
operations over $F$.

As explained in Section 3, since $U_{k+1}$ and $V_{k+1}$ are $A$-complementary, if $u$ and $v$ are uniformly, randomly and independently selected $S$-linear combinations of spanning sets of $U_{k+1}$ and $V_{k+1}$ respectively, then the probability that

$$\minpol(A^T, u) = \minpol(u^T, A, v) = \minpol(A, v) = \minpol(U_{k+1}^T, A, V_{k+1})$$

is at least $1 - \frac{2l}{|S|}$, where $l = \deg(\minpol(U_{k+1}^T, A, V_{k+1}))$.

One final observation about the Lanczos process (as described by Lambert [17]) can now be stated and used to good effect. This improves the complexity bounds stated in Facts 2.2 and 2.5 for the special case suggested above.

**Fact 4.4.** Let $A \in F^{n \times n}$ and let $u, v \in F^{n \times 1}$. If

$$\minpol(A^T, u) = \minpol(u^T, A, v) = \minpol(A, v)$$  (21)

then it is possible to compute the coefficients and degree $d$ of $\minpol(u^T, A, v)$ and dual bases for $A$, $u$ and $v$, deterministically, by computing the product of $A$ and $d$ vectors, $A^T$ and $d$ vectors, and performing $O(dn)$ operations over $F$. On the other hand, if condition (21) is not satisfied, then this can be detected deterministically at the same cost.

**Proof.** The above condition can be checked and values computed using a slightly modified version of Algorithm 3.5.1 of Lambert [17], in which one uses the vectors $u$ and $v$ as the input vectors $u_{\text{curr}}$ and $v_{\text{curr}}$ required by the algorithm, and in which one sets $b = 0$. It is necessary to add two additional comparisons — namely, to compare $u_{\text{curr}}$ to $v_{\text{curr}}$ whenever it is discovered that one of these equals zero, to check whether condition (21) is satisfied. However, since $b = 0$ and the vectors $y$ and $z$ are only used by this algorithm to try to generate a vector $z$ such that $A z = b$, it is also possible to reduce the cost of the computation by eliminating all statements involving either of these vectors.

The above complexity bounds now follow, for the case that condition (21) is satisfied, by a straightforward (and, conservative) analysis of the cost of the resulting algorithm. The bound can be achieved for the remaining case by keeping track of the number of operations used and reporting failure if this bound is reached before the algorithm would otherwise terminate. \(\square\)

A more careful inspection of Lambert’s Algorithm 3.5.1 suggests that at most $20dn + 5d^2 + O(n)$ additional operations over $F$ will be used. However, the less precise bound stated above will be sufficient.

Henceforth it will be assumed that a function $\text{fast-minpol-and-dual-basis}(A, u, v)$ returns the values mentioned in the above lemma if condition (21) is satisfied, reports failure if the condition is not satisfied, and can be implemented using a deterministic algorithm with the above complexity.

The function $\text{frobenius-large}$ shown in Figure 8 can be used to find the Frobenius form of a given matrix $A \in F^{n \times n}$ with entries in an infinite field $F$.

**Theorem 4.5.** Suppose $F$ is an infinite field, $A \in F^{n \times n}$, $\epsilon > 0$, and that $S$ is a finite subset of $F$ including at least $2n/\epsilon$ distinct elements.

If the function $\text{frobenius-large}$ is invoked on input $A$ and chooses field elements uniformly and independently from $S$, then the function returns the number $k$ of invariant factors of $A$, the invariant factors $f_1, f_2, \ldots, f_k$, and vectors $u_1, u_2, \ldots, u_k$ and $v_1, v_2, \ldots, v_k$ such that

$$\minpol(A^T, u_i) = \minpol(u_i^T, A, v_i) = \minpol(A, v_i) = f_i$$
function frobenius-large(A) begin
k := 0; d := 0
while d < n do
  k := k + 1
  u_k := randU-large(); v_k := randV-large()
  Apply fast-minpol-and-dual-basis(A, u_k, v_k), either to confirm that
  minpol(A^T, u_k) = minpol(u_k, A, v_k) = minpol(A, v_k)
  and to generate f_k = minpol(u_k^T, A, v_k), d_k = deg(f_k), and dual bases
  u_{k,1}, u_{k,2},..., u_{k,d_k} and v_{k,1}, v_{k,2},..., v_{k,d_k} for A, u_k and v_k, or to report
  failure. Terminate the computation and report failure if failure is
  reported in this step.
  if k > 1 and f_k does not divide f_{k-1} then
    report failure (and terminate the computation)
  else
    d := d + d_k
  end if
end while
return k; f_1, f_2,..., f_k; u_1, u_2,..., u_k; v_1, v_2,..., v_k end function

Figure 8: Function frobenius-large

for 1 ≤ i ≤ k, and such that u_i^T A^l v_j = 0 whenever 1 ≤ i, j ≤ k, i ≠ j, and l ≥ 0, with probability
at least 1 − ε (and reports failure, otherwise).

If A has k nontrivial invariant factors then the function can be implemented to use at most
n multiplications of A by vectors, at most n multiplications of A^T by vectors, and at most 8kn^2 +
O(n^2) additional operations over F.

Proof. Suppose A has nontrivial invariant factors f_1, f_2,..., f_k ∈ F[x] with degrees d_1, d_2,..., d_k
respectively, so that d_i is positive for all i and ∑_{i=1}^k d_i = n. Suppose, furthermore, that the first i−1
of these invariant factors have been computed correctly, where 1 ≤ i ≤ k.

Then, as noted in Section 3.1, a trivial modification of the argument given by Kaltofen and
Pan [14] to prove their Lemmas 1 and 2 can be used to establish that the likelihood that the
attempt to compute f_i fails is at most 2^k d_i / |S|. Therefore, the probability that the algorithm fails at
all can be bounded by ∑_{i=1}^k 2^d_i / |S| = 2n / |S| ≤ (2n)/(2n/ε) = ε, as claimed.

The claimed complexity bounds follow immediately from Lemma 4.3, Fact 4.4, and the obser-
vation that it is possible to check whether the i^{th} guessed invariant factor f_i divides the i−1^{st}
invariant factor f_{i-1} using O(deg(f_{i-1})^2) operations over F, so that the total cost of arithmetic
used by the algorithm, excluding that needed for invocations of randU-large, randV-large, or fast-
minpol-and-dual-basis, is in O(n^2).

39
Once again, a more careful count of the number of operations used could be made after a closer inspection of Lambert’s algorithm. This suggests that $8kn^2 + 27n^2 + 23kn + O(n)$ operations suffice.

As noted at the beginning of Section 4, a Frobenius transition matrix can be computed cheaply from this algorithm’s output, so that $2n$ multiplications of $A$ by vectors, $n$ multiplications of $A^T$ by vectors, and $8kn^2 + 27n^2 + 23kn + O(n)$ operations over $\mathbb{F}$ are sufficient to compute the Frobenius form and a Frobenius transition matrix for $A$, if $\mathbb{F}$ is sufficiently large.

4.3 An Asymptotically Fast Algorithm

Suppose now that $\mathbb{F}$ is an arbitrary field. An asymptotically fast version of the algorithm, such that the expected number of operations over $\mathbb{F}$ used by the algorithm is in $O(\mathcal{M}(n) \log n)$, under the common assumption that $\mathcal{M}(n) \in \Omega(n^{2+\epsilon})$ for a constant $\epsilon > 0$, will be presented next. This matches the asymptotic complexity bound established by Giesbrecht [12] for the large field case, and improves the known bounds for the case that $\mathbb{F}$ is small.

Organization of Data. Following a preprocessing stage, the algorithm will attempt to accumulate vectors whose Krylov spaces correspond to the blocks in the Frobenius form of the given matrix, and to generate dual bases for these spaces, like the previous algorithms — the data to be manipulated is as described at the beginning of Section 4.

However, it will be useful during the execution of the new algorithm to distinguish between those candidate invariant factors $f_i$, vectors $u_i$ and $v_i$, and dual bases that were generated before the last $\Theta(\log n)$ attempts to discover an invariant factor, and those invariant factors, vectors, and dual bases that have been generated more recently. We will say that the former invariant factors and associated data are committed. The latter are uncommitted, until it is discovered that all invariant factors have been computed. In order to decide which data are committed, the algorithm will keep track of the number $\text{time}$ of attempts to discover an invariant factor that have been made so far, and will associate a “time stamp” $\text{time\_stamp}_i$ with each uncommitted invariant factor $f_i$, that is set at the time of the first attempt to compute this factor. The invariant factor $f_i$ will become committed when $\text{time} - \text{time\_stamp}_i \geq 2 \log n$.

As usual the algorithm will keep track of the sum $d$ of the degrees of all invariant factors that have been discovered. The algorithm can terminate, with success, as soon as $d = n$, so all invariant factors will become “committed” once this condition is attained.

The algorithm will terminate, and fail, at any point when it is discovered that a committed invariant factor is incorrect. It will be shown below that the probability of failure is at most $1/n$.

In order to allow block matrix operations to be performed, it will be useful to maintain additional data as well: Recall that the algorithms described in this section all compute vectors $u_i, v_i \in \mathbb{F}^{n \times 1}$ such that

$$ \text{minpol}(A^T, u_i) = \text{minpol}(u_i^T, A, v_i) = \text{minpol}(A, v_i) = f_i $$

for each invariant factor $f_i$; the asymptotically fast algorithm to be described will also compute and use the vectors $(A^T)^j u_i$ and $A^j v_i$, for $1 \leq j < \deg(f_i)$.

Finally, additional data will be maintained in order to ensure that “random” vectors from various subspaces are always available. This data will be described as the algorithm is given in more detail.

Outline of Algorithm. The algorithm begins with a preprocessing stage, “Stage 0,” in which the matrix powers $A^{(2^i)}$ are computed, for $1 \leq i \leq \lfloor \log_2 n \rfloor$. These matrix powers will be maintained and used in the rest of the algorithm.
In the next stage, “Stage 1,” all invariant factors $f$ with degree greater than or equal to $n/\log n$ are computed and become committed.

The algorithm continues with at most $\log(n/\log n) = \log n - \log \log n$ additional stages, called “Stage $i + 1$,” for $i = 1, 2, \ldots, \lceil \log n - \log \log n \rceil$. By the end of Stage $i + 1$, all invariant factors $f$ with degree greater than or equal to $n/(2^i \log n)$ are computed and become committed.

For $i \geq 1$, Stage $i$ ends when either it is discovered that the sum $d$ of the degrees of all known invariant factors equals $n$ (in which case, the algorithm terminates successfully), failure is detected (because a “committed” invariant factor is discovered to be incorrect, in which case, the algorithm terminates unsuccessfully), or an invariant factor $f$ with degree less than $n/(2^i - 1) \log n$ becomes committed, in which case the algorithm proceeds to Stage $i + 1$.

**Preprocessing and its Effects.** As noted above, the algorithm will begin in Stage 0 by computing $A^{(2^i)}$ for $1 \leq i \leq \lfloor \log_2 n \rfloor$. This processing step can clearly be implemented at cost $O(\mathcal{MM}(n) \log n)$.

Since

$$A^{(2^i)} [w|Aw| \ldots |A^{2^i-1}w] = [A^{(2^i)}w|A^{2^i+1}w| \ldots |A^{2^{i+1}-1}w],$$

it is easily seen that the vectors $Aw, A^2w, \ldots, A^nw$ can be computed from the above powers of $A$ and from a given vector $w$ using $O(\mathcal{MM}(n))$ operations, if $\mathcal{MM}(n) \in \Omega(n^{2+\epsilon})$ for $\epsilon > 0$ — see Borodin and Munro [5], page 128, or Keller-Gehrig [16]. Furthermore, if $0 \leq h \leq \log n$ and $w_1, w_2, \ldots, w_{2^h} \in \mathbb{F}^{n \times 1}$, then it is also possible to compute the matrix-vector products $A^jw_i$ for $0 \leq j \leq \lfloor n/2^h \rfloor$ and $1 \leq i \leq 2^h$ at this cost. Clearly, since $(A^T)^{(2^i)} = (A^{(2^i)})^T$, one could also compute matrix-vector products $(A^T)^jw_i$ for $0 \leq j \leq \lfloor n/2^h \rfloor$ and $1 \leq i \leq 2^h$ at this cost, as well.

When these matrix powers and vectors are available, several of the computations that have previously discussed can be performed more efficiently, as summarized below.

**Evaluation of $f(A)w$ or $f(A^T)w$ for $f \in \mathbb{F}[x]$.** Clearly, if the coefficients of a polynomial $f \in \mathbb{F}[x]$ with degree less than $k$ are available, as well as the matrix-vector products $A^jw$ (respectively, $(A^T)^jw$) for $0 \leq j < k$, then $f(A)w$ (respectively, $f(A^T)w$) can be computed using $O(kn)$ operations over $\mathbb{F}$.

**Computation of the Minimum Polynomial of a Sequence.** Given $A$, $u$, $v$, an upper bound $k$ on the degree of $\text{minpol}(u^T, A, v)$, and the matrix-vector products $A^ju, (A^T)^2u, \ldots, (A^T)^ku$ and $Av, A^2v, \ldots, A^kv$, one can compute the sequence

$$u^Tv, u^TA^Tv, u^TA^2v, \ldots, u^TA^{2k-1}v,$$

by computing $2k$ inner products of vectors, and then apply the Berlekamp-Massey algorithm to generate the coefficients of $\text{minpol}(u^T, A, v)$, using $O(kn)$ operations over $\mathbb{F}$ (cf. Fact 2.2).

**Computation of a Dual Basis.** It will be necessary (and sufficient) to compute dual bases for a matrix $A$ and vectors $u$ and $v$ in the special case that

$$\text{minpol}(A^T, u) = \text{minpol}(u^T, A, v) = \text{minpol}(A, v)$$

and this polynomial (and its degree, $k$) is known. Assuming that $A^ju, (A^T)^2u, \ldots, (A^T)^{k-1}u$ and $Av, A^2v, \ldots, A^{k-1}v$ are also available, let $U \in \mathbb{F}^{k \times n}$ be the matrix with rows

$$u^T, u^TA^T = ((A^T)u)^T, u^TA^2 = ((A^T)^2u)^T, \ldots, u^TA^{k-1} = ((A^T)^{k-1}u)^T,$$
and let \( V \in F^{n \times k} \) be the matrix with columns

\[ v, A^v, A^2v, \ldots, A^{k-1}v. \]

Then the matrix \( T_v = UV \in F^{k \times k} \) is nonsingular and can be computed from the above values using \( O(\frac{n}{k} MM(k)) \) operations over \( F \). Its inverse, \( T_v^{-1} \), can then be computed using \( O(MM(k)) \) operations. Finally, since \( T_v^{-1}UV \) is the identity matrix of order \( k \), the columns of the matrices \( (T_v^{-1}U)^T \in F^{n \times k} \) and \( V \in F^{n \times k} \) form dual bases for \( A, u \) and \( k \), that can be computed from the given values using \( O(\frac{n}{k} MM(k)) \) operations over \( F \).

**Computation of the Minimum Polynomial of a Vector.** As noted in Section 2.2, the minimum polynomial \( \text{minpol}(A, v) \) of a matrix \( A \) and vector \( v \) can be computed, with high probability, as the least common multiple of a constant number of minimum polynomials \( \text{minpol}(u_i^T, A, v) \) of sequences, for independently selected vectors \( u_1, u_2, \ldots \). It follows that if \( \text{minpol}(A, v) \) has degree at most \( k \), and both the vectors \( Av, A^2v, \ldots, A^{k-1}v \) and \( A^Tu_i, (A^T)^2u_i, \ldots, (A^T)^{k-1}u_i \) have been precomputed for sufficiently many vectors \( u_i \), a Las Vegas algorithm can be used to compute \( \text{minpol}(A, v) \) using \( O(kn) \) operations over \( F \), with the algorithm always generating a factor of \( \text{minpol}(A, v) \) and correctly generating \( \text{minpol}(A, v) \) with probability at least one-half. If independent trials of this algorithm are performed and the least common multiple of the output polynomials is maintained, then it is clear that the expected number of operations that are performed, before \( \text{minpol}(A, v) \) is obtained, is in \( O(kn) \) as well.

**Computation of the Minimum Polynomial of a Subspace.** Now consider the problem of computing the minimum polynomial of a subspace, as discussed in Sections 2.3 and 3.1. Suppose, in particular, that \( U \) and \( V \) are \( A \)-complementary subspaces and that we wish to compute \( f = \text{minpol}(U^T, A, V) \), \( d = \deg(f) \), vectors \( u \in U \) and \( v \in V \) such that

\[ \text{minpol}(A^T, u) = \text{minpol}(u^T, A, v) = \text{minpol}(A, v) = f, \]

and the sequences of vectors

\[ u, A^Tu, (A^T)^2u, \ldots, (A^T)^{d-1}u \quad \text{and} \quad v, Av, A^2v, \ldots, A^{d-1}v. \]

These values can be generated using a version of the process described in Section 3.1 that uses asymptotically fast matrix arithmetic and that takes advantage of the precomputed matrix-vector products described above. The following lemma will be of use in describing and analyzing asymptotically fast variants of these algorithms.

**Lemma 4.6.** Let \( A \in F^{n \times n} \), \( v \in F^{n \times 1} \), and suppose \( f = \text{minpol}(A, v) \) and \( k = \deg(f) \). Given a polynomial \( g \in F[x] \) with degree less than \( k \), and given \( A, v, f, k \) and the sequence of vectors

\[ v, Av, A^2v, \ldots, A^{k-1}v, \]

it is possible to compute the sequence of vectors

\[ g(A)v, Ag(A)v, A^2g(A)v, \ldots, A^{k-1}g(A)v \]

using \( O(\frac{n}{k} MM(k)) \) operations over \( F \).
Proof. Since the degree of \( g \) is less than that of \( f \), the coefficients of \( g \mod f = g \) are given. The coefficients of \( g^{i+1} \mod f \) can be computed from those of \( g^i \mod f \) using \( O(k) \) operations over \( F \) for any nonnegative integer \( i \) using standard polynomial arithmetic, so the coefficients of the polynomials
\[
g \mod f, \ xg \mod f, \ x^2g \mod f, \ldots, \ x^{k-1}g \mod f
\]
can be computed using \( O(k^2) \) operations with standard polynomial arithmetic.

Now suppose
\[
g^i \mod f = g_{i,0} + g_{i,1}x + \cdots + g_{i,k-1}x^{k-1} \in F[x]
\]
where \( g_{i,j} \in F \) for \( 0 \leq i, j \leq k - 1 \), and let
\[
M_v = \begin{bmatrix} v & Av & \ldots & A^{k-1}v \end{bmatrix} \in F^{n \times k} \quad \text{and} \quad M_g = \begin{bmatrix} g_{0,0} & g_{1,0} & \cdots & g_{k-1,0} \\ g_{0,1} & g_{1,1} & \cdots & g_{k-1,1} \\ \vdots & \vdots & \ddots & \vdots \\ g_{0,k-1} & g_{1,k-1} & \cdots & g_{k-1,k-1} \end{bmatrix} \in F^{k \times k}.
\]
Then the entries of these matrices are available after the above polynomials have been computed and, since \( f = \text{minpol}(A,v) \), the matrix \( M_v,M_g \in F^{n \times k} \) has the desired vectors
\[
g(A)v, A\hat{g}(A)v, \ldots, A^{k-1}g(A)v
\]
as its columns. Since this matrix can be computed from \( M_v \) and \( M_g \) using \( O(n^2MM(k)) \) operations over \( F \) and \( k^2 \in O(n^2MM(k)) \) as well, the desired result now follows.

It can now be established that the desired values can be computed for a given pair of \( A \)-complementary subspaces \( U \) and \( V \) by considering each of the algorithms given in Section 3.1: Suppose, once again, that the degree of \( \text{minpol}(U^T,A,V) \) is at most \( k \). Then, if \( \hat{u} \in U \) and \( \hat{v} \in V \) and, furthermore, the vectors
\[
\hat{u}, A^T\hat{u}, (A^T)^2\hat{u}, \ldots, (A^T)^{k-1}\hat{u} \quad \text{and} \quad \hat{v}, A\hat{v}, A^2\hat{v}, \ldots, A^{k-1}\hat{v}
\]
are available, then a version of the function \text{filterv} that calls the function \text{filterp} and asymptotically fast versions of \text{minpolseq} and \text{minpolvec} can be used to generate vectors \( u \in U \) and \( v \in V \) with the properties (and relationship to \( \hat{u} \) and \( \hat{v} \)) described in Figure 2 and Lemma 3.2, either using \( O(n^2MM(k)) \) operations over \( F \) in the worst case and failing with probability less than one-half or, using independent trials, with an expected complexity as above. By the above lemma, the vectors \( u, A^Tu, (A^T)^2u, \ldots, (A^T)^{k-1}u \) and \( v, A^Tv, A^2v, \ldots, A^{k-1}v \) could also be computed at this cost. The function \text{mergev} shown in Figure 3 and discussed in Lemma 3.3 can also be extended so that its outputs include vectors \( u, A^Tu, (A^T)^2u, \ldots, (A^T)^{k-1}u \) and \( v, A^Tv, A^2v, \ldots, A^{k-1}v \) and executed at this cost. Finally, then, if vectors
\[
\hat{u}_i, A^T\hat{u}_i, (A^T)^2\hat{u}_i, \ldots, (A^T)^{k-1}\hat{u}_i \quad \text{and} \quad \hat{v}_i, A\hat{v}_i, A^2\hat{v}_i, \ldots, A^{k-1}\hat{v}_i
\]
have been precomputed for a constant number of vectors \( \hat{u}_1, \hat{u}_2, \cdots \in U \) and \( \hat{v}_1, \hat{v}_2, \cdots \in V \), then the function \text{minpolspace} shown in Figure 4 and discussed in Theorem 3.4 can be extended to generate vectors \( u, A^Tu, (A^T)^2u, \ldots, (A^T)^{k-1}u \) and \( v, A^Tv, A^2v, \ldots, A^{k-1}v \) along the vectors \( u \) and \( v \), and either to use the above number of operations in the worst case and fail with probability one-half, or (using independent trials) to generate correct output after using an expected number of operations as above.

43
Implementation and Cost of Stage 1  Stage 1 can be implemented using the steps shown in the while-loop in the algorithm for the small field case shown in Figure 6, with the addition of code needed to maintain the time stamps described at the beginning of this section, and using the asymptotically fast versions of subroutines that have been described above.

An inspection of the above material will confirm that each attempt to discover a new invariant factor can be performed at cost $O(\mathcal{M}(n))$. Since the expected number of attempts needed to find each factor is bounded by a constant, and there are at most $\log n$ invariant factors of any matrix $A \in \mathbb{F}^{n \times n}$ with degree greater than or equal to $n/\log n$, the expected number of operations that are performed before each of these factors — and the first invariant factor with smaller degree — has been found is in $O(\mathcal{M}(n) \log n)$. The number of attempts that must be performed after that, before the last of these factors is committed, is at most $2 \log n$ as well, so the total cost of all operations performed in Stage 1 is in $O(\mathcal{M}(n) \log n)$ as needed.

Implementation of Cost of Stage $i + 1$ for $i \geq 1$ An implementation of Stage $i + 1$ will now be described and analyzed, for $i \geq 1$, in order to show that all but one part of this stage can be implemented correctly using $O(\mathcal{M}(n))$ operations, and that the total cost needed to execute the final part for all stages is in $O(\mathcal{M}(n) \log n)$. Since there are fewer than $\log n$ of these stages, it will follow that the total cost of the algorithm is in $O(\mathcal{M}(n) \log n)$, as promised.

Recall that the purpose of Stage $i + 1$ is to discover and commit all the invariant factors with degree at most $n/2^i \log n$. Clearly, at most $2^i \log n$ such factors exist.

The most significant change in the implementation of this stage concerns the choice of “random” vectors. In order to make this efficient, vectors will be chosen and processed in blocks of size $2^i \log n$, so that the expected number of blocks whose vectors should be processed before all the desired invariant factors are found and committed is bounded by a constant.

Orthogonalization of Vectors.  Let $m_i = [2^i \log n]$ and let $b_i = [n/m_i]$. A block of “random” vectors will be generated by selecting vectors $x_1, x_2, \ldots, x_{b_i}$ and $v_1, v_2, \ldots, v_{m_i}$ independently and randomly from $\mathbb{F}^{n \times n}$, and then using the dual bases for subspaces corresponding to the invariant factors found so far to orthogonalize these vectors, as well as vectors of the form $(A^T)^j w_h$ and $A^i x_h$ for $1 \leq j \leq b_i$ and $1 \leq h \leq m_i$, with respect to these subspaces.

The next lemma and its corollary are needed to prove that this orthogonalization process is correct.

Lemma 4.7. Let $U$ and $V$ be $A$-complementary subspaces of $\mathbb{F}^{n \times 1}$ and suppose $u_1, u_2, \ldots, u_m$ and $v_1, v_2, \ldots, v_m$ are dual bases for $A$, $U$, and $V$. Let $v \in \mathbb{F}^{n \times 1}$ and let $i$ be a nonnegative integer. Finally, let

$$v^\perp = v - \sum_{j=1}^{m} \frac{u_j^T v}{u_j^T u_j} v_j \quad \text{and} \quad (A^i v)^\perp = A^i v - \sum_{j=1}^{m} \frac{u_j^T (A^i v)}{u_j^T u_j} v_j.$$  

Then $A^i(v^\perp) = (A^i v)^\perp$.

Proof. It is clear by the definition of $v^\perp$ that (since $v_1, v_2, \ldots, v_m \in V$)

$$v - v^\perp = \sum_{j=1}^{m} \frac{u_j^T v}{u_j^T u_j} v_j \in V.$$  

Since $V$ is $A$-invariant, $A^i v - A^i(v^\perp) = A^i(v - v^\perp) \in V$ as well. Since it is also clear by the definition of $(A^i v)^\perp$ that $A^i v - (A^i v)^\perp \in V$,

$$A^i(v^\perp) - (A^i v)^\perp = \left(A^i v - (A^i v)^\perp\right) - \left(A^i v - A^i(v^\perp)\right) \in V.$$  

44
On the other hand, if $1 \leq k \leq m$ then it follows by the definition of $v^\perp$ that
\[
    u_k^T v^\perp = u_k^T v - u_k^T \sum_{j=1}^{m} \frac{u_j^T v}{u_j^T v_j} v_j
    = u_k^T v - \sum_{j=1}^{m} \frac{u_j^T v}{u_j^T v_j} u_k^T v_j
    = u_k^T v - \frac{u_k^T v}{u_k^T v_k} u_k^T v_k = 0,
\]
since $u_1, u_2, \ldots, u_m$ and $v_1, v_2, \ldots, v_m$ are dual bases for $A$, $U$ and $V$. Therefore (again, since $u_1, u_2, \ldots, u_m$ is a basis for $U$) $u^T v^\perp = 0$ for all $u \in U$ and, since $U$ is $A^T$-invariant, it follows that if $1 \leq k \leq m$ then
\[
    u_k^T (A^t(v^\perp)) = ((A^T)^i u_k)^T v^\perp = 0
\]
as well. It also follows by the definition of $(A^t v)^\perp$ that $u_k^T ((A^t v)^\perp) = 0$ and, therefore, that
\[
    u_k^T ((A^t(p)^\perp) - (A^t v)^\perp) = u_k^T (A^t(v^\perp)) - u_k^T (A^t v)^\perp = 0
\]
for $1 \leq k \leq m$. However, $(A^t(v)^\perp) - (A^t v)^\perp \in V$ as established above and, since $u_1, u_2, \ldots, u_m$ and $v_1, v_2, \ldots, v_m$ are dual bases for $A$, $U$ and $V$, this implies that $A^t(v^\perp) - (A^t v)^\perp = 0$, as required.

The following corollary follows from the fact that if $U$ and $V$ are $A$-complementary then $V$ and $U$ are $A^T$-complementary.

**Corollary 4.8.** Let $U$ and $V$ be $A$-complementary subspaces of $F^{n \times 1}$ and suppose $u_1, u_2, \ldots, u_m$ and $v_1, v_2, \ldots, v_m$ are dual bases for $A$, $U$ and $V$. Let $u \in F^{n \times 1}$ and let $i$ be a nonnegative integer. Finally, let
\[
    u^\perp = u - \sum_{j=1}^{m} \frac{u_j^T v_j}{u_j^T v_j} u_j \quad \text{and} \quad ((A^T)^i u)^\perp = (A^T)^i u - \sum_{j=1}^{m} \frac{((A^T)^i u)^T v_j}{u_j^T v_j} u_j.
\]
Then $(A^T)^i (u^\perp) = ((A^T)^i u)^\perp$.

Now, a block of random vectors will be generated from $w_1, w_2, \ldots, w_m$, and $x_1, x_2, \ldots, x_m$, by using the following steps, after choosing $w_1, w_2, \ldots, w_m$, and $x_1, x_2, \ldots, x_m$, uniformly and independently from $F^{n \times 1}$, or uniformly and independently from $S^{n \times 1}$ for a sufficiently large finite subset $S$ of $F$.

1. Orthogonalize these vectors with respect to vector spaces corresponding to all invariant factors that are committed before this generation of this block of vectors begins.
2. Orthogonalize vectors (as they are needed) with respect to vector spaces corresponding to all invariant factors that became committed after the generation of this block of vectors began.
3. Orthogonalize vectors (as they are needed) with respect to vector spaces corresponding to all uncommitted invariant factors.
The first two orthogonalization steps can be performed using $O(\mathcal{M}(n))$ operations per stage, and the total cost of the third orthogonalization step for all stages is in $O(\mathcal{M}(n) \log n)$.

To perform the first step, one should begin by computing the powers $(A^T)^jw_h$ and $A^jx_h$ for $1 \leq j \leq b_i$ and $1 \leq h \leq m_i$. Let

$$ W = [w_1 \quad A^Tw_1 \quad \ldots \quad (A^T)^{b_i}w_1 \quad \ldots \quad w_{m_i} \quad A^Tw_{m_i} \quad \ldots \quad (A^T)^{b_i}w_{m_i}] \in \mathbb{F}^{n \times m_i(b_i+1)} $$

and let

$$ X = [x_1 \quad Ax_1 \quad \ldots \quad A^{b_i}x_1 \quad \ldots \quad x_{m_i} \quad Ax_{m_i} \quad \ldots \quad A^{b_i}x_{m_i}] \in \mathbb{F}^{n \times m_i(b_i+1)}. $$

Furthermore, let $k_c$ be the number of committed invariant factors discovered before this orthogonalization process begins and let $d_c = d_1 + d_2 + \cdots + d_{k_c}$ be the sum of the degrees of all these committed invariant factors. Let $U \in \mathbb{F}^{n \times d_c}$ and $V \in \mathbb{F}^{n \times d_c}$ be matrices whose columns are vectors in the bases for the $A^T$-invariant and $A$-invariant subspaces, respectively, chosen from the dual bases corresponding to the these committed invariant factors. That is,

$$ U = [u_{1,1} \quad u_{1,2} \quad \ldots \quad u_{1,d_1} \quad u_{k_c,1} \quad u_{k_c,2} \quad \ldots \quad u_{k_c,d_{k_c}}] $$

and

$$ V = [v_{1,1} \quad v_{1,2} \quad \ldots \quad v_{1,d_1} \quad v_{k_c,1} \quad v_{k_c,2} \quad \ldots \quad v_{k_c,d_{k_c}}]. $$

Then $U^TV = V^TU \in \mathbb{F}_d \times d_e$ is a nonsingular diagonal matrix, by the construction of these dual bases. Finally, suppose $W$ and $X$ are modified as follows.

$$ W := W - U \cdot (V^TU)^{-1}(V^TW) \quad \text{and} \quad X := X - V \cdot (U^TV)^{-1}(U^TX). $$

Then it follows by Lemma 4.7 and Corollary 4.8 that, following these operations,

$$ W = [w_1 \quad A^Tw_1 \quad \ldots \quad (A^T)^{b_i}w_1 \quad \ldots \quad w_{m_i} \quad A^Tw_{m_i} \quad \ldots \quad (A^T)^{b_i}w_{m_i}] $$

and

$$ X = [x_1 \quad Ax_1 \quad \ldots \quad A^{b_i}x_1 \quad \ldots \quad x_{m_i} \quad Ax_{m_i} \quad \ldots \quad A^{b_i}x_{m_i}], $$

where $w_1, w_2, \ldots, w_{m_i}$ are now randomly selected vectors from the subspace orthogonal to the vectors $v_{1,1}, v_{1,2}, \ldots, v_{k_c,d_e}$ and $x_1, x_2, \ldots, x_{m_i}$ are randomly selected vectors orthogonal to the vectors $u_{1,1}, u_{1,2}, \ldots, u_{d_e,k_c}$, as desired.

Since $m_ib_i < 2n$ and $m_i(b_i+1) < 3n$, and $d_c \leq n$, this part of the orthogonalization process can be executed using $O(\mathcal{M}(n))$ operations over $\mathbb{F}$, as claimed.

The second part of the orthogonalization process is complicated by the fact that it involves invariant factors that are uncommitted (and unknown, or likely to change) when the process begins. This process will be interleaved with the use of the vectors it generates to discover additional invariant factors.

Let $j$ be an integer between 1 and $m_i$, and set $h_j$ and $e_j$ to be zero, if no invariant factor becomes committed immediately after $w_j$ and $x_j$ are used. Otherwise, let $h_j$ be the index of the invariant factor (“$f_{h_j}$”) that becomes committed at this time, and let $e_j = d_{h_j}$, the degree of this invariant factor.

Set $U_j$ and $V_j$ to be “empty” matrices (with $n$ rows and zero columns) if $h_j = 0$. Otherwise, let

$$ U_j = [u_{h_j,1} \quad u_{h_j,2} \quad \ldots \quad u_{h_j,e_j}] \quad \text{and} \quad V_j = [v_{h_j,1} \quad v_{h_j,2} \quad \ldots \quad v_{h_j,e_j}] $$

be matrices in $\mathbb{F}^{n \times e_j}$ have the vectors in the dual bases corresponding to this invariant factor as its columns.
procedure orthog-new(low, high)

Input: Integers low, high such that 1 ≤ low ≤ high ≤ m_i

begin procedure

if low < high then
    mid := [(low + high)/2]
    orthog-new(low, mid - 1)
    W_{mid,high} := W_{mid,high} - U_{low, mid-1} \cdot (V_{low, mid-1}^T \cdot U_{low, mid-1})^{-1} \cdot V_{low, mid-1}^T \cdot W_{mid,high}
    X_{mid,high} := X_{mid,high} + V_{low, mid-1} \cdot (U_{low, mid-1}^T \cdot V_{low, mid-1})^{-1} \cdot U_{low, mid-1}^T \cdot X_{mid,high}
    orthog-new(mid, high)
else if high < m_i then suspend end if
end if
end function

Figure 9: On-the-Fly Orthogonalization for Recently Committed Invariant Factors

For all pairs of integers \( h, j \) such that \( 1 \leq h \leq j \leq m_i \), let \( e_{h,j} = e_h + e_{h+1} + \cdots + e_j \),

\[
U_{h,j} = \begin{bmatrix} U_h & U_{h+1} & \cdots & U_j \end{bmatrix} \quad \text{and} \quad V_{h,j} = \begin{bmatrix} V_h & V_{h+1} & \cdots & V_j \end{bmatrix}
\]

so that \( U_{h,j}, V_{h,j} \in \mathbb{F}^{n \times e_h} \). Similarly, let

\[
W_j = \begin{bmatrix} w_j & (A^T)^{h_{1}} w_j & \cdots & (A^T)^{h_j} w_j \end{bmatrix} \quad \text{and} \quad X_j = \begin{bmatrix} x_j & Ax_j & \cdots & A^{h_j} x_j \end{bmatrix}
\]

and let

\[
W_{h,j} = \begin{bmatrix} W_h & W_{h+1} & \cdots & W_j \end{bmatrix} \quad \text{and} \quad X_{h,j} = \begin{bmatrix} X_h & X_{h+1} & \cdots & X_j \end{bmatrix}
\]

so that \( W_{h,j}, X_{h,j} \in \mathbb{F}^{n \times (j-h+1)(b_i+1)} \), \( W = W_{1,m_i} \), and \( X = X_{1,m_i} \).

Vectors will be orthogonalized with respect to dual bases for newly committed invariant factors using a process orthog-new that is shown in Figure 9. This process will be invoked with inputs 1 and \( m_i \), immediately after vectors \( w_1 \) and \( x_1 \) have been used to try to discover a new invariant factor. The code shown in Figure 9 includes a “suspend” operation, which passes control back to a process that uses a pair of vectors to find a new invariant factor. It is easily proved by induction on \( i \) that each pair of vectors \( w_i \) and \( x_i \) will be orthogonalized with respect to all invariant factors that become committed when \( w_1, x_1, w_2, x_2, \ldots, w_{i-1}, x_{i-1} \) were used, by the time \( w_i \) and \( x_i \) are to be used themselves, as long the process that generates invariant factors passes control back to the orthogonalization process (perhaps, by executing a resume operation) after each attempt to find an invariant factor.

It is easily checked, by inspection of the code, that if \( T(low, high) \) is the number of operations used by this procedure on inputs low and high, then

\[
T(low, high) \leq T(low, mid - 1) + T(mid, high) + c \frac{n}{(high - low + 1)(b_i + 1)} \mathcal{MM}((high - low + 1)(b_i + 1))
\]

47
\( s := l + h \)
\( \text{found} := \text{false} \)

while \( s \geq l - 1 \) and not \( \text{found} \) do

Orthogonalize \( w_i, A^T w_i, \ldots, (A^T)^{d_i} w_i \) and \( x_i, A x_i, \ldots, A^{d_i} x_i \) with respect to the dual spaces for all uncommitted invariant factors.

if \( f_s(A^T)w_i = 0 \) and \( f_s(A)x_i = 0 \) then

\( \text{found} := \text{true}; \ k := d_s \)

else

\( s := s - 1 \)

end if

end while

Figure 10: Orthogonalization for Uncommitted Invariant Factors
expected number of operations to orthogonalize all vectors with respect to all invariant factors can now be bounded by
\[ O\left( n \log n \sum d_j \right) = O(\mathcal{M}(n) \log n), \]
as required. Thus, the total cost needed to orthogonalize vectors is in \( O(\mathcal{M}(n) \log n) \).

The algorithm that has been sketched can fail with small probability. Clearly, an algorithm that never fails, returns the desired values, and terminates with probability zero, can be obtained by performing independent trials of the above algorithm until a trial is successful. The above analysis implies the following.

**Theorem 4.9.** Let \( A \in \mathbb{F}^{n \times n} \). Then the number \( k \) of invariant factors of \( A \), the invariant factors \( f_1, f_2, \ldots, f_k \), and vectors \( u_1, u_2, \ldots, u_k \) and \( v_1, v_2, \ldots, v_k \) such that
\[
\minpol(A^T u_i) = \minpol(u_i^T, A, v_i) = \minpol(A, v_i) = f_i
\]
for \( 1 \leq i \leq k \) and such that \( u_i^T A^l v_j = 0 \) whenever \( 1 \leq i, j \leq k, i \neq j, \) and \( l \geq 0 \), can be computed by a Las Vegas algorithm that terminates with probability one, using an expected number of operations in \( \mathbb{F} \) in \( O(\mathcal{M}(n) \log n) \).

### 4.4 Inverting the Frobenius Transition Matrix

Consider again a matrix \( A \in \mathbb{F}^{n \times n} \) and the values generated by the above algorithms, including the degrees \( d_1, d_2, \ldots, d_k \) of the nontrivial invariant factors \( f_1, f_2, \ldots, f_k \) of this matrix, and vectors \( u_1, u_2, \ldots, u_k \) and \( v_1, v_2, \ldots, v_k \) such that
\[
\minpol(A^T u_i) = \minpol(u_i^T, A, v_i) = \minpol(A, v_i) = f_i
\]
for \( 1 \leq i \leq k \) and such that \( u_i^T A^l v_j = 0 \) whenever \( 1 \leq i, j \leq k, i \neq j, \) and \( l \geq 0 \). As previously noted, a Frobenius transition matrix \( V \) with columns
\[ v_1, A v_1, \ldots, A^{d_i - 1} v_1, \ldots, v_k, A v_k, \ldots, A^{d_k - 1} v_k \]
can be computed from these values using at most \( n - 1 \) additional multiplications of \( A \) by vectors. Furthermore, a matrix \( U \) with columns
\[ u_1, A^T u_1, \ldots, (A^T)^{d_i - 1} u_1, \ldots, u_k, A^T u_k, \ldots, (A^T)^{d_k - 1} u_k \]
can also be computed using at most \( n - 1 \) additional multiplications of \( A^T \) by vectors. The field elements
\[ u_i^T v_1, u_i^T A v_1, \ldots, u_i^T A^{2d_i - 2} v_i \]
can then be computed for all \( i \) using \( O(n^2) \) additional operations, by computing \( O(n) = O(\sum_{i=1}^k d_i) \) inner products of vectors.

Now, note once again that
\[
U^T V = H = \begin{bmatrix}
H_1 & 0 \\
H_2 & \ddots \\
0 & \cdots & H_k
\end{bmatrix}
\]
where

\[ H_i = \begin{bmatrix}
    u_i^T v_i & u_i^T A v_i & \cdots & u_i^T A^{d_i-1} v_i \\
    u_i^T A v_i & u_i^T A^2 v_i & \cdots & u_i^T A^{d_i} v_i \\
    \vdots & \vdots & \ddots & \vdots \\
    u_i^T A^{d_i-1} v_i & u_i^T A^{d_i} v_i & \cdots & u_i^T A^{2d_i-2} v_i
\end{bmatrix} \in \mathbb{F}^{d_i \times d_i},
\]

so that \( H_i \) is a nonsingular Hankel matrix for \( 1 \leq i \leq k \) and \( H \) is a nonsingular block diagonal matrix with Hankel matrices on its blocks.

Clearly, then \( V^{-1} = H^{-1} U^T \). If the field \( \mathbb{F} \) supports a fast Fourier transform, then the entries of the matrix \( V^{-1} \) can be computed by solving \( n \) block-Hankel systems of linear equations, using \( O(n^2 \log^2 n) \) operations over \( \mathbb{F} \) — see Brent, Gustavson and Yun [6] for details.

Alternatively, if one simply wishes to solve a single system \( Vx = y \) for a given vector \( y \), then this can be accomplished using \( O(n^2) \) operations over \( \mathbb{F} \) — the computation of the vector \( U^T y \) (before solving the system \( Hx = U^T y \)) will dominate the cost. One can also obtain a representation of \( V^{-1} \) as a product of two matrices at the cost of computing \( H^{-1} \). The entries of this matrix can be generated from \( H^{-1} \) using \( O(n \log^2 n) \) operations if a fast Fourier transform is available, and they can always be generated using \( O(n^2) \) operations — see Bini and Pan [4] for a discussion of Hankel matrix inversion and additional references.

5 Computation of a Rational Jordan Form

It is also known (see, again, Gantmacher [9]) that every matrix \( A \in \mathbb{F}^{n \times n} \) is similar to a block-diagonal matrix

\[
J_A = \begin{bmatrix}
    J_{g_1} & 0 \\
    J_{g_2} & \ddots \\
    0 & \ddots & J_{g_l}
\end{bmatrix}
\]

(22)

for distinct, monic, irreducible polynomials \( g_1, g_2, \ldots, g_l \in \mathbb{F}[x] \), where each block \( J_{g_i} \) is block-diagonal with companion matrices on its blocks:

\[
J_{g_i} = \begin{bmatrix}
    C_{g_i}^{e_{i,1}} & C_{g_i}^{e_{i,2}} & \cdots & 0 \\
    0 & \ddots & \ddots & \cdots \\
    \vdots & \ddots & \ddots & \ddots \\
    0 & \cdots & \cdots & C_{g_i}^{e_{i,m_i}}
\end{bmatrix},
\]

(23)

where \( m_i > 0 \) and \( e_{i,1} \geq e_{i,2} \geq \cdots \geq e_{i,m_i} > 0 \) for \( 1 \leq i \leq l \). The polynomials \( g_1, g_2, \ldots, g_l \) are unique (up to the order in which they are listed), and the matrix \( J_A \) is unique, up to the order of the diagonal blocks. Every such matrix \( J_A \) is called a rational Jordan form of \( A \), and every nonsingular matrix \( W \in \mathbb{F}^{n \times n} \) such that

\[
W A W^{-1} = J_A
\]

is called a Jordan transition matrix for \( A \).

We will now show that a rational Jordan form and a Jordan transition matrix for a matrix \( A \in \mathbb{F}^{n \times n} \) can be computed efficiently from \( A \), if the field \( \mathbb{F} \) is finite. Suppose, for the rest of this section, that the Frobenius form \( F_A \), a Frobenius transition matrix \( V \), associated matrix \( U \), and the
block diagonal matrix \( H = U^T V \) (with Hankel blocks on the diagonal) described in the previous section are available.

Let \( F(n) \) be the expected number of arithmetic operations over \( F \) required, in the worst case, to factor a polynomial \( f \) of degree \( n \) in \( F[x] \). If the algorithm of Berlekamp [3] is used, and \( F \) is a finite field of size \( q \), one can take \( F(n) \in O(MM(n) + n^2 \log q) \). Asymptotically faster algorithms also exist; in particular, if one uses the algorithm of Kaltofen and Shoup [15], one can take \( F(n) \in O(n^{1.515} \log q) \). See Kaltofen and Shoup [15] as well for additional references concerning factorization of polynomials over finite fields.

**Lemma 5.1.** Given a matrix \( A \in \mathbb{F}^{n \times n} \) and the Frobenius form of \( A \), a rational Jordan form of \( A \) can be computed using an expected number of operations over \( F \) in \( O(n^2) + F(n) \).

**Proof.** The invariant factors \( f_1, f_2, \ldots, f_k \) of \( A \) are available, since the Frobenius form of \( A \) has the companion matrices of these polynomials as its blocks.

It is well known that if the rational Jordan form of \( A \) is as given above, in Equations (22) and (23), then \( k = \max(m_1, m_2, \ldots, m_l) \) and the \( j \)th invariant factor \( f_j \) of \( A \) has factorization

\[
f_j = g_1^{e_{1,j}} g_2^{e_{2,j}} \cdots g_l^{e_{l,j}}
\]

in \( F[x] \), for \( 1 \leq j \leq k \), with \( e_{i,j} = 0 \) whenever \( j > m_l \). Since the rational Jordan form has the companion matrices of the powers of irreducible polynomials \( g_i^{e_{i,j}} \) on its blocks, for \( 1 \leq i \leq l \) and \( 1 \leq j \leq m_i \), a rational Jordan form for \( A \) can clearly be written down using \( O(n^2) \) steps, once the coefficients of these polynomials are known.

The polynomials \( g_1, g_2, \ldots, g_l \) and exponents \( e_{i,1}, e_{2,1}, \ldots, e_{l,1} \) can be computed using an expected number of operations in \( O(\mathcal{F}(n)) \) by factoring the minimum polynomial \( f_1 \) of \( A \). Let \( d_i \) be the degree of \( g_i \) for \( 1 \leq i \leq l \); then, since \( d_i e_{i,1} + d_{2} e_{2,1} + \cdots + d_l e_{l,1} \) is the degree of \( f_j \) and is at most \( n \), the polynomials \( g_1^{e_{1,j}}, g_2^{e_{2,j}}, \ldots, g_l^{e_{l,j}} \) can clearly be computed using \( O(n^2) \) operations over \( F \), using repeated squaring, with standard polynomial arithmetic.

Suppose, now, that \( 2 \leq j \leq k \); then, for \( 1 \leq i \leq l \),

\[
g_i^{e_{i,j}} = \gcd(g_i^{e_{i,j-1}}, f_j),
\]

since \( e_{i,j} \geq e_{i,j-1} \) and \( g_i^{e_{i,j}} \) is the largest power of \( g_i \) dividing \( f_j \). Thus, \( g_i^{e_{i,j}} \) can be computed from \( g_i^{e_{i,j-1}} \) and \( f_j \), with standard polynomial arithmetic, using a number of operations that is linear in the product of the degree \( d_i e_{i,j-1} \) of \( g_i^{e_{i,j-1}} \) and the degree of \( f_j \). The total number of operations needed to compute \( g_1^{e_{1,j}}, g_2^{e_{2,j}}, \ldots, g_l^{e_{l,j}} \) from \( g_1^{e_{1,j-1}}, g_2^{e_{2,j-1}}, \ldots, g_l^{e_{l,j-1}} \) and \( f_j \) is therefore at most linear in the product of the degrees of \( f_{j-1} \) and \( f_j \). This is clearly at most linear in the product of \( n \) and \( f_j \). Therefore, since the sum of the degrees of \( f_1, f_2, \ldots, f_k \) is \( n \), the total number of operations over \( F \) required to compute \( g_{i,j} \) for \( 1 \leq i \leq l \) and \( 2 \leq j \leq m_i \), from \( g_{i,1}, g_{i,2}, \ldots, g_{i,i} \), is at most quadratic in \( n \), as required.

It is also possible to generate both a Jordan transition matrix \( W \) for \( A \) and its inverse \( W^{-1} \), each as a product of three matrices: Since \( J_A \) is similar to \( A \), \( J_A \) has the same Frobenius form \( F_A \) for \( A \). Let \( \hat{V} \) be a Frobenius transition matrix for \( J_A \), and let \( \hat{U} \) be the corresponding matrix that could also be generated using the algorithms described in earlier sections, so that \( \hat{U}^T \hat{V} = \hat{H} \) is a block diagonal matrix with Hankel blocks; then

\[
VAV^{-1} = \hat{V} J_A \hat{V}^{-1} = F_A,
\]

so that \( WAW^{-1} = J_A \) if \( W = \hat{V}^{-1} V = \hat{H}^{-1} \hat{U}^T V \), so that \( W^{-1} = V^{-1} \hat{V} = H^{-1} U^T \hat{V} \). A proof of the following result is now straightforward.
Theorem 5.2. Let \( A \in \mathbb{F}^{n \times n} \) be a matrix over a field \( \mathbb{F} \). The rational Jordan form \( J_A \) of \( A \), and matrices \( U, V, H', \hat{U}, \hat{V}, \) and \( \hat{H}' \in \mathbb{F}^{n \times n} \) such that \( W = \hat{H}'\hat{U}^T\hat{V} \) is a Jordan transition matrix for \( A \) and \( W^{-1} = H'U^TV \), can be computed at an expected cost of \( O(n) \) multiplications of \( A \) by vectors, \( O(n) \) multiplications of \( A^T \) by vectors, and \( O(kn^2 + \mathcal{F}(n)) \) operations over \( \mathbb{F} \), where \( k \) is the number of invariant factors of \( A \).

Proof. The Frobenius form of \( A \), a Frobenius transition matrix \( V \), and corresponding matrices \( U \) and \( H \) such that \( U^TV = H \) is block diagonal with Hankel blocks, can all be computed at the above cost as described in Section 4. As noted in Section 4.4, the matrix \( H' = H^{-1} \) can be computed from \( H \) using \( O(n^2) \) additional operations over \( \mathbb{F} \).

Now, a rational Jordan form \( J_A \) of \( A \) can be computed from \( A \) and \( F_A \) at the above cost as well, by Lemma 5.1. Since \( J_A \) is similar to \( A \) it also has Frobenius form \( F_A \). Since it is block diagonal with companion matrices as blocks, it has at most \( 2n \) nonzero entries, so that either \( J_Ax \) or \( J_A^Tx \) can be computed from \( J_A \) and a given vector \( x \) using \( O(n) \) operations over \( \mathbb{F} \). It follows that a Frobenius transition matrix \( \hat{V} \) for \( J_A \), and corresponding matrices \( \hat{U} \) and \( \hat{H} \) such that \( \hat{U}^T\hat{V} = \hat{H} \) is also block diagonal with Hankel blocks, can be computed using \( O(kn^2) \) operations over \( \mathbb{F} \). The matrix \( \hat{H}' = \hat{H}^{-1} \) can be computed using an additional \( O(n^2) \) operations. Now \( W = \hat{H}'\hat{U}^T\hat{V} = \hat{H}^{-1}\hat{U}^T\hat{V} \) is a Jordan transition matrix for \( A \) and \( W^{-1} = H'U^TV = H^{-1}U^TV \) as explained above.

Using an asymptotically fast Frobenius form algorithm as given in Section 4.3, and using asymptotically fast matrix multiplication to compute the entries of a rational Jordan form (from the three matrices of which it is a product), one can also establish the following using the same outline.

Theorem 5.3. Let \( A \in \mathbb{F}^{n \times n} \). A rational Jordan form \( J_A \) and a Jordan transition matrix \( W \) such that \( WAW^{-1} = J_A \) can be computed using a Las Vegas algorithm at an expected cost of \( O(MM(n) \log n + \mathcal{F}(n)) \) operations over \( \mathbb{F} \).

6 Acknowledgements

Gabor Ivanyos suggested that there was work left to do on this problem in the small field case, and has my thanks.

Discussions with Mark Giesbrecht, Erich Kaltofen, David Saunders, Gilles Villard, and other members of the Linbox project concerning black box linear algebra, and the relationship between the Lanczos and Wiedemann algorithms, were extremely helpful.

References


