On the computation of minimal polynomials, cyclic vectors, and Frobenius forms

Daniel Augot, Paul Camion

ON THE COMPUTATION OF MINIMAL POLYNOMIAL, CYCLIC VECTORS AND THE FROBENIUS FORM

DANIEL AUGOT, PAUL CAMION

Abstract. Various algorithms connected with the computation of the minimal polynomial of a square $n \times n$ matrix over a field k are presented here. The complexity of the first algorithm, where the complete factorization of the characteristic polynomial is needed, is $O(\sqrt{n}n^3)$. It produces the minimal polynomial and all characteristic subspaces of a matrix of size n. Furthermore an iterative algorithm for the minimal polynomial is presented with complexity $O(n^3+n^2m^2)$, where m is a parameter of the used Shift-Hessenberg matrix. It does not require knowledge of the characteristic polynomial. Important here is the fact that the average value of m or $m_{\mathbf{A}}$ is $\approx \log n$.

Next we are concerned with the topic of finding a cyclic vector first for a matrix whose characteristic polynomial is square-free. Using the Shift-Hessenberg form leads to an algorithm at cost $O(n^3 + m^2n^2)$. A more sophisticated recurrent procedure gives the result in $O(n^3)$ steps. In particular, a normal basis for an extended finite field will be obtained complexity $O(n^3 + n^2 \log q)$.

Finally the Frobenius form is obtained with asymptotic average complexity $O(n^3 \log n)$. All algorithms are deterministic. In all four cases, the complexity obtained is better than for the heretofore best known deterministic algorithm. The results are summarized in Tables 1, 2, 3 and 4.

1. Introduction

We present various low complexity algorithms for computing the objects in the title.

Section 2 is concerned with the problem of obtaining the minimal polynomial of a square matrix A. The algorithm introduced recquires the factorization of the characteristic polynomial of A for its input, and produces the minimal polynomial and all characteristic subspaces

Key words and phrases. characteristic polynomial, Hessenberg form, characteristic subspace, minimal polynomial, cyclic vector, normal basis, Frobenius form, centralizer of a matrix.

Problem	Complexity	Average complexity	Section
Minimal polynomial	$O(n^{3.5})$	$O(n^{3.5})$	2
Minimal polynomial	$O(n^3 + n^2 m_A^2)$	$O(n^3)$	4
Cyclic vector	$O(n^3 + n^2 m_A^2)$	$O(n^3)$	5
Cyclic vector	$O(n^3)$	$O(n^3)$	6
Normal basis	$O(n^3 + n^2 \log q)$	$O(n^3 + n^2 \log q)$	6
Frobenius form	$O(n^3 m_A)$	$O(n^3 \log n)$	7

TABLE 1. Summary of complexities

at cost $O(\sqrt{n}n^3)$. It appeals to a recurrent "divide-and-conquer" procedure.

The Section 3 is of theoretical nature, we introduce the Shift-Hessenberg form of a matrix, whose algebraic properties are studied. Asymptotic are also derived, from the results of R. Stong [8]. Basic algorithmic with that form is detailed.

In Section 4, using this form, we obtain an iterative algorithm ending in the minimal polynomial in $O(n^3 + n^2 m^2)$ elementary operations over $\mathbb{F}_q$. It does not need any knowledge of the characteristic polynomial. The number m is a parameter of the Shift-Hessenberg form, and we have that $m \leq m_{\mathbf{A}}$, where m_A is the number of factors of the characteristic polynomial of A, counted with multiplicities.

We next are concerned with the topic of finding a cyclic vector. We will construct an algorithm for matrices whose characteristic polynomial is square-free. Under that assumption, the Shift-Hessenberg form leads to an algorithm of complexity $O(n^3 + m^2 n^2)$ presented in Section 5, and to a more sophisticated recurrent procedure with complexity $O(n^3)$ presented in Section 6. Special attention is given to cyclic vectors for the Frobenius automorphism of a finite field. This ends in a deterministic algorithm for computing a normal basis for $\mathbb{F}_{q}^{\kappa}$ in $O(n^3 + n^2 \log q)$ elementary operations. This algorithm is better than previously known deterministic algorithms [2], and is comparable to the probabilistic algorithms presented in [9, 5].

In Section 7, the Frobenius form is obtained with asymptotic average complexity $O(n^3 \log n)$.

The results are summarized in Table 1. We shall compare the complexities of our algorithms to the algorithms described in P. Ozello's work [6]. Ozello's methods have been chosen for implementation by the authors of the Maple system.

Note 1. *Our complexity measures are given in terms of elementary operations over k. All algorithms presented here may be applied to*

matrices over any field $\mathbf{k}$ *and in particular over* $\mathbb{Q}$*, but we don't give any measure of the bit-complexity.*

2. Characteristic subspaces and minimal polynomial in $O(n^{3.5})$

In this section, an algorithm with complexity $O(n^3\sqrt{n})$ is presented for computing the minimal polynomial of a matrix A, and a block-diagonal matrix D similar to A and exhibiting its characteristic subspaces. The inputs are A and the factorization of its characteristic polynomial. The output are the minimal polynomial, a block-diagonal matrix D exhibiting the restriction of A at its characteristic subspaces, and an invertible matrix P such that $D = P^{-1}AP$.

Note that the characteristic polynomial of a matrix A can be computed in $O(n^3)$ elementary operations, as described in [11, 3], by computing a *Hessenberg* form of A. The characteristic polynomial can be factored at low cost, for instance $O(n^3 + n^3 \log q)$ [10], although it is not a deterministic algorithm.

2.1. Characteristic subspaces.

We recall known facts about characteristic subspaces of a matrix A. The reader may refer to [4]. Let $\mathbf{k}$ be a field, we denote $M_n(\mathbf{k})$ the algebra of $n \times n$ matrices, and $GL_n(\mathbf{K})$ the group of invertible $n \times n$ matrices.

Theorem 1. *Let $C(X)$ be the characteristic polynomial of matrix $A \in M_n(\mathbf{k})$, and assume $C(X) = P(X)Q(X)$ where $P(X)$ and $Q(X)$ are relatively prime. Let $V_P = \ker P(A)$ and $V_Q = \ker Q(A)$, then*

$$\mathbf{k}^n = V_P \oplus V_Q, \text{ and } V_P = \text{Im } Q(A) \text{ and } V_Q = \text{Im } P(A).$$

Definition 1. *Let $C(X)$ be the characteristic polynomial of matrix A, and let $C(X)$ factors into $f_1(X)^{r_1} \cdots f_k(X)^{r_k}$, where the polynomials f_i are irreducible. The* characteristic subspaces *of A are the invariant subspaces $V_i = \ker f_i(A)^{r_i}$, $i = 1, \ldots, k$.*

2.2. The algorithm.

The strategy of the algorithm is as follows. If the characteristic polynomial of A is $C(X) = p(X)^r$ where $p(X)$ is irreducible, then $\mathbf{k}^n$ is a characteristic subspace, and finding the minimal polynomial of A reduces to finding the minimal exponent s such that $p(A)^s = 0$.

If the characteristic polynomial is not a power of an irreducible polynomial, we are able to split $C(X)$ into $C(X) = P(X)Q(X)$ with $P(X)$ and $Q(X)$ relatively prime and either $P(X)$ or $Q(X)$ is of degree greater than $\frac{2}{3}n$ and is a power of an irreducible polynomial, or we have that $\deg P(X), \deg Q(X) \leq \frac{2}{3}n$. We recursively apply the procedure on both

V_P and V_Q, given by Theorem 1. The new matrices are split in their turn, until all characteristic subspaces of A are obtained. Finally the minimal polynomial of the restriction of A to each of those subspaces is computed. The product of those polynomials gives the final result.
Inputs: matrix A and the factorization of its characteristic polynomial, $C(X) = f_1(X)^{r_1} \dots f_k(X)^{r_k}$, where $f_1(X), \dots, f_k(X)$ are the irreducible factors of $C(X)$.
Outputs: the minimal polynomial of A, the splitting of $\mathbf{k}^n$ into all characteristic subspaces of A, and the matrix of the restriction A to the characteristic subspaces.
Step 1: Find a splitting of $C(X) = P(X)Q(X)$ where $P(X)$ and $Q(X)$ are coprime. Three cases are considered.

- $C(X) = p(X)^r$, $p(X)$ irreducible. Compute the minimal polynomial $p(X)^s$ of A in $\lceil \log_2 r \rceil$ steps by trial and error on s. This is done with complexity $O(n^3\sqrt{n})$, using Theorem 2, which follows.
- One factor, $p_i(X)^{r_i}$, has degree larger than $\frac{2}{3}n$. Then $P(X) = p_i(X)^{r_i}$, i.e. $C(X) = p_i(X)^{r_i}Q(X)$, and $Q(A)$ gives a basis for a characteristic subspace.
- All factors $p_i(X)^{r_i}$ have degree $\leq \frac{2}{3}n$. Find a splitting $C(X) = P(X)Q(X)$ where $P(X)$ and $Q(X)$ are relatively prime and where $\deg P(X) \leq \frac{2}{3}n$ and $\deg Q(X) \leq \frac{2}{3}n$. This is described in Lemma 1, which follows.

Step 2: Compute $Q(A)$, $P(A)$. This gives generating vectors for subspaces for V_P and V_Q respectively. This is done at cost $O(n^3\sqrt{n})$, using Theorem 2.
Step 3: Compute bases for V_P and V_Q respectively. This is done with Gauss elimination, at cost $O(n^3)$.
Step 4: Change basis, taking for the new basis the union of the bases just computed, compute the matrices A_P and A_Q of the restriction of A to V_P and V_Q respectively. The cost is again $O(n^3)$.
Recursive Step Recursively apply the procedure to A_P and A_Q, terminal steps end in basis for all characteristic subspaces by giving the diagonal blocks of D.

Now two main operations are to be performed. The splitting and the evaluation of the polynomials $P(X)$ and $Q(X)$ at A, with complexity $O(n^3\sqrt{n})$, are detailed in next section.

2.3. **Splitting the factors, and evaluation.** We state the following useful lemma.

Lemma 1. *Let n and $n_i, i \in [1,k]$ be positive integers such that $n_1 + \cdots + n_k = n$, and $n_i \leq \frac{2}{3}n$, for $1 \leq i \leq k$. Then there exists a partition $[1,k] = I \cup J$ such that:*

$$n_I \leq \tfrac{2}{3}n, \text{ and } n_J \leq \tfrac{2}{3}n.$$

Proof. For every subset $J \subset [1,k]$, denote S_J the sum $\sum_{i \in J} n_J$. If there exists $n_i > \frac{1}{3}n$, choose $I = \{i\}$ and $J = [1,k] \setminus I$.

Otherwise, choose J as the subset of $[1.k]$ of maximal size such that $S_j \leq \frac{2}{3}n$. Then $I = [1,k] \setminus J$ necessary satisfies $S_i \leq \frac{2}{3}n$. Indeed, if $S_i > \frac{2}{3}n$, let I' be constructed from I by removing any of its element. Then $S_{i'} > \frac{2}{3}n - \frac{1}{3}n$, since $n_i \leq \frac{1}{3}n, i \in [1,k]$. Then the complementary J' of I' in $[1,k]$ satisfies $S_{J'} \leq \frac{2}{3}n$ and contains J. This contradicts the maximality of J. □

Thus, given k integers $n_1, \ldots, n_k$ summing up to n, we are able to find a splitting $[1,k] = I \cup J$ such that either $I = \{i\}$, and $i > \frac{2}{3}n$, either both sets I and J are such that $n_I \leq \frac{2}{3}n$ and $n_J \leq \frac{2}{3}n$.

We now recall how $p(A)$ can be computed at cost $\sqrt{t}n^3$, where t is the degree of $p(X)$. A naïve Horner algorithm would lead to $O(tn^3)$. This result has been shown in [7], and we recall it for completeness.

Theorem 2. *For all A in $M_n(\mathbf{k})$, for all $p(X)$ with $\deg p(X)$ at most t, $p(A)$ can be computed with complexity $O(\sqrt{t}n^3)$, and memory space of $O(\sqrt{t}n^2)$.*

2.4. **The complexity.**

Theorem 3. *Given the factorization of its characteristic polynomial, the minimal polynomial of any square matrix over a finite field $\mathbf{k}$ and a block-diagonal matrix similar to A exhibiting its characteristic sub-spaces, is computed with time complexity $O(n^3\sqrt{n})$, and memory size $O(n^2\sqrt{n})$.*

Proof. We can assume that all intermediate computations for splitting the factors, re-evaluating matrices, and computing bases for sub-spaces are all bounded by $\gamma n^{3.5}$. We have to show that the whole recursive algorithm has complexity bounded by $O(n^{3.5})$. We prove it by recurrence, assuming that the cost $C(m)$ of the algorithm is bounded by $\beta m^{3.5}$, for $m < n$. Then:

$$C(n) \leq \gamma n^{3.5} + 2C(\tfrac{2}{3}n) \leq \gamma n^{3.5} + 2\beta \left(\tfrac{2}{3}\right)^{3.5} n^{3.5}.$$

Thus $C(n) \leq \beta n^{3.5}$, with $\beta = \frac{\gamma}{1-2(\frac{2}{3})^{3.5}}$. □

3. The Shift-Hessenberg form and the centralizer of a matrix

We now use the Shift-Hessenberg form of a matrix. The main point is that evaluating a polynomial at a matrix is less expensive when that matrix has the Shift-Hessenberg form. The average improvement is, as will be seen, considerable. Before going to the use of the Shift-Hessenberg form for our algorithmic purposes, we show how Shift-Hessenberg forms shed light on the subgroup of $GL_n(\mathbf{k})$ commuting with a given fixed linear operator on $\mathbf{k}^n$.

3.1. **Shift-basis.**

Let A be a square matrix, we denote $\pi(A)$ the minimal polynomial of A.

Definition 2. *For A in $M_n(\mathbf{k})$ and v in $\mathbf{k}^n$, the* minimal polynomial of A restricted to v *is the lowest degree monic non-zero polynomial $\pi_v(X)$ such that $\pi_v(A)v = 0$.*

Definition 3. *Let A be an operator on $\mathbf{k}^n$. A* shift-basis *for A is a basis which has the form:*

$$\left[v_1, Av_1, \dots, A^{n_1-1}v_1, v_2, Av_2, \dots, A^{n_2-1}v_2, \dots, v_m, Av_m, \dots, A^{n_m-1}v_m\right], \tag{1}$$

such that $A^{n_k}v_k$ is linearly dependent of the A^iv_j, $j < k$, and of the A^iv_k, $i < n_k$, and, for $l < n_k$, A^lv_k is linearly independent of the A^iv_k, $i < l$, and of the A^iv_j, $j < k$.

It is understood that a shift-basis is actually an *ordered basis*. Given A, a shift-basis for A can be obtained as follows. First select any v_1, and introduce the linear independent vectors A^iv_1, for $i = 0, \dots, n_1-1$, where n_1 is the smallest value of i such that A^iv_1 linearly depends on all previous vectors. Then select v_2 independent of all previous vectors, and proceed with A^iv_2, $i = 0, \dots, n_2 - 1$ as for v_1. The process ends in a shift-basis with $n_1 + n_2 + \dots + n_m = n$

Definition 4. *We call a matrix which represents an operator A in a shift-basis a* Shift-Hessenberg *matrix. A Shift-Hessenberg matrix has*

the following form:

$$H = \begin{bmatrix} & & \times & & & & \times & & & \times \\ 1 & & \times & & & & \times & & & \times \\ & 1 & \times & & & & \times & & & \times \\ & & 0 & & & & \times & & & \times \\ & & & & 1 & & \times & & & \times \\ & & & & & \ddots & \times & & & \\ & & & & & & 1 & \times & & \times \\ & & & & & & & 0 & & \times \\ & & & & & & & & 1 & \times \\ & & & & & & & & & \ddots \end{bmatrix}.$$

The parameter *m of a Shift-Hessenberg matrix is defined to be the number of zeros on the first subdiagonal, plus one.*

Note 2. *The number m is the number of diagonal blocks, each block being a companion matrix, i.e. a matrix of the form:*

$$\begin{bmatrix} & & & c_0 \\ 1 & & & c_1 \\ & \ddots & & c_i \\ & & 1 & c_{n-1} \end{bmatrix}.$$

The characteristic polynomial of such a matrix equals its minimal polynomial and is equal to $X^n - c_{n-1}X^{n-1} - c_{n-2}X^{n-1} \cdots - c_1X^1 - c_0$. In the case where the parameter $m = 1$, the Shift-Hessenberg matrix is itself a companion matrix. We shall denote C_p the companion matrix associated to the polynomial p. The other extreme situation is for $m = n$, where we have an upper triangular matrix.

To every shift-basis corresponds an increasing sequence $V_1 \subset \ldots \subset V_m$ of invariant subspaces of A, defined by $V_1 = \text{Vect}\{A^i v_1; i \geq 0\}$, $V_2 = \text{Vect}\{A^i v_1, A^j v_2; i, j \geq 0\}$, $\ldots$, $V_m = \mathbf{k}^n$.

We have that V_i is a $\mathbf{k}[A]$-module, $i = 1, \ldots, m$ and consequently V_i/V_{i-1} is a module . Each of those quotient modules is generated by a single element $\overline{v}_i$ which is the class of v_i in V_i/V_{i-1}. If the minimal polynomial of $\overline{v}_i$ is denoted by $f_i(X)$, the *ith* diagonal block in the Shift-Hessenberg matrix is the companion matrix of $f_i(X)$. We also note that $f_i(X)$ divides the minimal polynomial of v_i.

3.2. The $\mathbf{k}[X]$-module induced by a matrix.

Notation 1. *Let p be a polynomial and v a vector, we use the module notation pv for $p(A)v$. We use the same notation pv when p is a*

polynomial $p(A_V)$ evaluated at the restriction A_V of A an invariant subspace V, and when v is a vector in V.

Definition 5. *The Expanded-Frobenius form of A in $M_n(\mathbf{k})$ is the following matrix D similar to A:*

$$D = \begin{bmatrix} F_{B_1,B_1} & 0 & \cdots & 0 \\ 0 & F_{B_2,B_2} & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & F_{B_d,B_d} \end{bmatrix};$$

where each matrix F_{B_i,B_i} is a Frobenius matrix:

$$\begin{bmatrix} C_{p_i^{s_{i,1}}} & 0 & \cdots & 0 \\ 0 & C_{p_i^{s_{i,2}}} & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & C_{p_i^{s_{i,m_i}}} \end{bmatrix},$$

with $s_{i,1} \leq s_{i,2} \leq \ldots \leq s_{i,m_i}$, and with $\gcd(p_i, p_j) = 1$ if $i \neq j$.

We thus have that $p_i^{s_{i,m_i}}$ is the minimal polynomial of F_{B_i,B_i}. The subspaces for which the matrix is a companion matrix are denoted by $V_{p_i^{s_{i,1}}}, \ldots V_{p_i^{s_{i,m_i}}}$. The whole space $\mathbf{k}^n$ can be viewed as the direct sum $\bigoplus_{i=1}^{d} \bigoplus_{j=1}^{m_i} V_{p_i^{s_{i,j}}}$.

We consider $\mathbf{k}^n$ equipped with the natural structure of $\mathbf{k}[X]$-module induced by A. Then the module $\mathbf{k}^n$ can be represented as the product of rings:

$$R_{1,1} \times \cdots \times R_{1,m_1} \times R_{2,1} \times \cdots \times R_{2,m_2} \times \cdots \times R_{d,1} \times \cdots \times R_{d,m_d},$$

considered as $\mathbf{k}[X]$-modules where $R_{i,j} = \mathbf{k}[X]/p_i^{s_{i,j}}$. For any vector u, we denote by $u_{|R_{i,j}}$ the component of u in the ring $R_{i,j}$. Thus $u_{|R_{i,j}}$ may be considered indiscriminately as a vector or a polynomial of degree less than $s_{i,j}deg(p_i)$.

3.3. **Shift-bases for the Expanded-Frobenius form.** We study the nature of shift-bases which yield the expanded Frobenius form of an operator A.

Lemma 2. *Let u be a vector in $\mathbf{k}^n$, such that $p_i^{s_{i,j}} u = 0$. Then the components of u viewed in the $\mathbf{k}[X]$-module decomposition of $\mathbf{k}^n$ satisfy $u_{|R_{k,l}} = 0$ if $k \neq i$.*

Proof. Suppose there exists k, l, $k \neq i$ such that $u_{|R_{k,l}} \neq 0$. Then $p_i^{s_{i,j}} u_{|R_{k,l}}$ cannot be zero, since $u_{|R_{k,l}}$ is not zero, and since $p_i^{s_{i,j}}$ is a unit of $R_{k,l} = \mathbf{k}[X]/p_k^{s_{k,l}}$. This contradicts the assumption on u. □

Lemma 3. *Let u be a vector in $\mathbf{k}^n$, whose minimal polynomial is $p_i^{s_{i,j}}$. Then the components of u in $R_{i,l}$ are described as follows:*

(1) *$l < j$; $u_{|R_{i,l}}$ can be any element of $R_{i,l}$,*
(2) *$l = j$; $u_{|R_{i,l}}$, considered as a polynomial, is prime to p_i,*
(3) *$l > j$; $u_{|R_{i,l}}$ is a multiple of $p_i^{s_{i,u}-s_{i,j}}$.*

Proof. Since the minimal polynomial of u is $p_i^{s_{i,j}}$, then we have that $p_i^{s_{i,j}} v = 0$ for any vector v in $R_{i,l}$, whenever $l < j$, since $p_i^{s_{i,l}}$, which divides $p_i^{s_{i,j}}$, is the minimal polynomial of A restricted to $R_{i,l}$. This establishes the result for the case $l < j$.

In case $l = j$, a vector is cyclic for a companion matrix if, considered as a polynomial, it is relatively prime to the minimal polynomial of that matrix.

In case $l > j$ we must have that $p_i^{s_{i,j}} u = 0$. This implies in $R_{i,j}$ that $p_i^{s_{i,j}} u_{|R_{i,l}} = 0 \bmod p_i^{s_{i,l}}$, and thus we must have that $p_i^{s_{i,l}-s_{i,j}}$ divides $u_{|R_{i,l}}$. □

Property 1. *From the previous lemmas, all shift-bases yield the expanded Frobenius matrix D as described in definition 5 have the form:*

$$v_{1,1}, Dv_{1,1}, \ldots, D^{n_{1,1}-1}v_{1,1}, \ldots, v_{d,m_d}, Dv_{d,m_d}, \ldots, D^{n_{d,m_d}-1}v_{d,m_d},$$

where $n_{i,j} = s_{i,j} \deg p_i$, and where each $v_{i,j}$ is such that $p_i^{s_{i,j}}$ is its minimal polynomial.

3.4. **From shift-bases to the centralizer of a matrix.** Let $A \in M_n(\mathbf{k})$ be similar to the matrix S, with $S = P^{-1}AP$. Suppose that P' is another basis yielding the same form S of A, that is, $S = P'^{-1}AP'$. From the equality $P^{-1}AP = P'^{-1}AP'$, we get $(P'P^{-1})A = A(P'P^{-1})$. Thus $P'P^{-1}$ belongs to the centralizer of A. Conversely, let C belongs to the centralizer of A, and let $S = P^{-1}AP$. Since $AC = CA$, we get $S = (CP)^{-1}A(CP)$, thus CP is another basis yielding the matrix S for A.

In particular, if S is the expanded Frobenius form of A, then shift-bases yielding S are in correspondence with the elements of the centralizer of A. Suppose S has the form:

$$\begin{bmatrix} s & 0 \\ 0 & t \end{bmatrix},$$

where s is the restriction of A to V_s, t the restriction of A to V_t, with $V_s \oplus V_t = \mathbf{k}^n$. Suppose also that the minimal polynomials of s and t are coprime.

Let $v_{1,1} \ldots v_{k_s,m_{k_s}}$ be vectors producing a shift-basis for s, as in property 1, and let $p_i^{s_{i,j}}$ be the minimal polynomial of $v_{i,j}$. Let $v_{i,j}^\star$ denote the

vector $(v_{i,j}, 0) \in \mathbf{k}^n$, then the minimal polynomial of $v^\star_{i,j}$ is $p_i^{s_{i,j}}$. Similarly, let $w_{1,1} \dots w_{k_t,m_{k_t}}$ yield a shift-basis for t, with minimal polynomials $q_i^{r_{i,j}}$, and let $w^\star_{i,j}$ be the vector $(0, w_{i,j})$, whose minimal polynomial is also $q_i^{r_{i,j}}$. Then, by property 1, the family $v_{1,1} \dots v_{k_s,m_{k_s}}, w_{1,1} \dots w_{k_t,m_{k_t}}$ yields a shift-basis for A. We have proved:

Corollary 1. *The centralizer of the direct sum of two matrices s and t whose minimal polynomials are relatively prime is the direct product of the centralizers of s and t respectively.*

In the case where $\mathbf{k} = \mathbb{F}_q$ we can derive the enumeration of the centralizer of any given matrix.

Theorem 4. *Let A be an operator whose Expanded-Frobenius form is as in definition 5. The number of shift-basis for A which yield the above Frobenius form is*

$$\prod_{i=1}^{d} \prod_{j=1}^{m_i} q^{\deg(p_i)\left(\sum_{w=1}^{j-1} s_{i,w} + (m_i - j) s_{i,j}\right)} \phi(p_i^{s_{i,j}}). \tag{2}$$

where $\phi(p_i^{s_{i,j}}) = q^{s_{i,j} \deg(p_i)}(1 - q^{-\deg(p_i)})$ is the number of polynomials of degree less than $\deg(p_i^{s_{i,j}})$ prime to $p_i^{s_{i,j}}$.

Proof. Each such shift-basis is given by a sequence as:

$$v_{1,1}, v_{1,2}, \dots, v_{1,m_1}, v_{2,1}, \dots, v_{2,m_2}, \dots, v_{d,1} \dots v_{d,m_d},$$

in which the minimum degree polynomial canceling $v_{i,j}$ is $p_i^{s_{i,j}}$.

In formula (2), the outermost product is due to Lemma 2. The innermost product enumerates for each $p_i^{s_{i,j}}$ the number of vectors v such that $p_i^{s_{i,j}} v = 0$. The sum $\sum_{w=1}^{j-1} s_{i,w}$ stands for the rings $R_{i,l}$, $l < j$, in which any vector v satisfies $p_i^{s_{i,j}} v = 0$. The term $(m_i - j)s_{i,j}$ is a result of the fact that for every $l > j$ the number of polynomials multiple of $p_i^{s_{i,l} - s_{i,j}}$ in $\mathbf{k}[X]/p_i^{s_{i,l}}$ is $q^{\deg(p_i) s_{i,j}}$.

Finally, $\phi(p_i^{s_{i,j}})$ is the number of polynomials prime to $p_i^{s_{i,j}}$, i.e. the number of units in $R_{i,j}$. □

3.5. **The average number of factors of a characteristic polynomial.** R. Stong gives in [8] the following result.

Theorem 5. *Let X_n be the random variable assuming as values the number of factors of the characteristic polynomials of matrices in $GL_n(\mathbb{F}_q)$, counted with multiplicities. Then the expectation EX_n of X_n is asymptotically equivalent to $\log n$.*

We shall generalize the result to all matrices by proving the following:

Theorem 6. *Let Y_n be the random variable assuming as values the number of factors of characteristic polynomials of matrices in $M_n(\mathbb{F}_q)$, counted with multiplicities, and let EY_n be the expectation of Y_n. Then, for every $\epsilon > 0$, there exits n_0 such that $EY_n \leq 2(1+\epsilon)\log n$ for $n \geq n_0$.*

The proof of the Theorem needs two lemmas that will be first established. For any matrix $A \in M_n(\mathbb{F}_q)$, we consider its Expanded-Frobenius form as follows:

$$\begin{bmatrix} s & 0 \\ 0 & t \end{bmatrix}, \tag{3}$$

where s is a Frobenius form with characteristic polynomial X^{n_1} for some n_1, and t is an invertible matrix of size $n_2 = n - n_1$.

Lemma 4. *The average number EZ_n of factors counted with multiplicities of the characteristic polynomial of t, as in (3) , for matrices A in $M_n(\mathbb{F}_q)$, satisfies: $\forall \epsilon > 0, \exists n_0,\ n \geq n_0 \Rightarrow EZ_n \leq (1+\epsilon)\log n$.*

Proof. Let S_{n_1} be the set of Frobenius matrices with characteristic polynomial X^{n_1} and let S_{n_2} be the set of invertible Frobenius matrices whose characteristic polynomial has degree n_2. We denote by z_{s,n_1} the size of the centralizer of $s \in S_{n_1}$ and by z_{t,n_2} the size of the centralizer of $t \in S_{n_2}$. Given s in S_{n_1} and t in S_{n_2}, then by Corollary 1, the number of matrices having Frobenius form as in equation (3) is:

$$\frac{|GL_n(\mathbb{F}_q)|}{z_{s,n_1} z_{t,n_2}}.$$

The number of matrices having X^{n_1} in the decomposition of their characteristic polynomial and a fixed matrix t in their second diagonal block as in (3) is:

$$\sum_{s \in S_{n_1}} \frac{|GL_n(\mathbb{F}_q)|}{z_{s,n_1} z_{t,n_2}} = \frac{|GL_n(\mathbb{F}_q)|}{z_{t,n_2}} \sum_{s \in S_{n_1}} \frac{1}{z_{s,n_1}} = \frac{1}{z_{t,n_2}} \chi(n_1, n, q),$$

where $\chi(n_1, n, q) = |GL_n(\mathbb{F}_q)| \sum_{s \in S_{n_1}} \frac{1}{z_{s_{n_1}}}$.

Now let $C_{n_2,k}$ be the set of polynomials $C(X)$, $C(0) \neq 0$, of degree n_2 that split into k factors counted with multiplicities, and let $S_{n_2,k}$ be the set Frobenius matrices of size n_2 whose characteristic polynomial belongs to $C_{n_2,k}$. The number of matrices in $M_n(\mathbb{F}_q)$ whose characteristic polynomial is $X^{n_1}C(X)$, for $C(X)$ in $C_{n_2,k}$, is:

$$\chi(n_1, n, q) \sum_{t \in S_{n_2,k}} \frac{1}{z_{t,n_2}}.$$

Denote by θ the random variable assuming as value the size of the non-singular part of a matrix, and denote by η the random variable

assuming as value the number of factors of the characteristic polynomial of the non-singular part. The conditional probability $P_n\{\eta = k \mid \theta = n_2\}$ that $C(X)$ belongs to $C_{n_2,k}$ for a matrix in $M_n(\mathbb{F}_q)$ whose characteristic polynomial is $X^{n_1}C(X)$, is thus:

$$\frac{\chi(n_1,n,q)\sum\limits_{t\in S_{n_2,k}}\frac{1}{z_{t,n_2}}}{\chi(n_1,n,q)\sum\limits_{t\in S_{n_2}}\frac{1}{z_{t,n_2}}} = \frac{\sum\limits_{t\in S_{n_2,k}}\frac{|GL_{n_2}(\mathbb{F}_q)|}{z_{t,n_2}}}{\sum\limits_{t\in S_{n_2}}\frac{|GL_{n_2}(\mathbb{F}_q)|}{z_{t,n_2}}} = P_{n_2}\{\eta = k\},$$

where $P_n\{\eta = k\}$ denotes the probability that an invertible matrix in $GL(n,\mathbb{F}_q)$ has a characteristic polynomial which splits into k factors.

Now we can conclude: the expected number of factors of the invertible block of any matrix in $M_n(\mathbb{F}_q)$ is given by:

$$\begin{aligned}\sum_{k=1}^{n} k \sum_{n_2=1}^{n} P\{\theta = n_2\}P_{n_2}\{\eta = k\} &= \sum_{n_2=1}^{n} P\{\theta = n_2\}\sum_{k=1}^{n} kP_{n_2}\{\eta = k\}\\ &= \sum_{n_2=1}^{n} P\{\theta = n_2\}EX_{n_2}.\end{aligned}$$

Let ϵ be given. Since $EX_n \sim \log n$, there exists n_1 such that for $n \geq n_1$ then $EX_n/\log n \leq 1 + \epsilon/2$. Thus:

$$\begin{aligned}\frac{\sum_{n_2=1}^{n} P\{\theta = n_2\}EX_{n_2}}{\log n} &= \frac{\sum_{n_2=1}^{n_1} P\{\theta = n_2\}EX_{n_2}}{\log n} + \sum_{n_2=n_1+1}^{n} P\{\theta = n_2\}\frac{EX_{n_2}}{\log n}\\ &\leq \frac{\sum_{n_2=1}^{n_1} EX_{n_2}}{\log n} + \sum_{n_2=n_1+1}^{n} P\{\theta = n_2\}(1+\frac{\epsilon}{2})\\ &\leq \frac{\sum_{n_2=1}^{n_1} EX_{n_2}}{\log n} + 1 + \frac{\epsilon}{2}.\end{aligned}$$

We can choose n_0 such that, for all $n \geq n_0$, $\frac{EZ_n}{\log n} \leq 1 + \frac{\epsilon}{2}$. □

The proof of Theorem 6 will be completed by the following Lemma.

Lemma 5. *Let Z_n be the random variable assuming as values the number of factors X of characteristic polynomials of matrices in $M_n(\mathbb{F}_q)$. Then the expectation EZ_n is asymptotically bounded by* $\log n$.

Proof. Let us consider the translation $M \mapsto M + I_n$. The factor X^{n_1} of a matrix M becomes $(X-1)^{n_1}$ in the factorization of the characteristic polynomial of $M' = M + I_n$. Consider the Frobenius form of M':

$$\begin{bmatrix} s & 0 \\ 0 & t \end{bmatrix},$$

with s nilpotent and t invertible, then $(X-1)^{n_1}$ is the largest power of $X-1$ which is a factor of the characteristic polynomial $C(X)$ of t. By Lemma 4 the expected number of factors of $C(X)$ is asymptotically $\log n$, thus n_1 is asymptotically bounded by $\log n$. □

Theorem 6 now follows from Lemma 4 and Lemma 5.

3.6. **Computing with the Shift-Hessenberg form.** We recall the following theorem, whose proof and corresponding algorithm may be found in Ozello's thesis [6].

Theorem 7. *For all A in $M_n(\mathbf{k})$, there exists a Shift-Hessenberg matrix H and an invertible matrix P such that $H = PAP^{-1}$. The matrices H and P can be obtained in $O(n^3)$ elementary operations.*

We now investigate more precisely the number m of diagonal blocks of the Shift-Hessenberg form for a matrix A.

Definition 6. *Let A be a square matrix in $M_n(\mathbf{k})$. We denote by m_A the maximum size of an increasing sequence $V_1 \subset \ldots \subset V_{m_A}$ of invariant subspaces of $\mathbf{k}^n$ under A.*

Since any Shift-Hessenberg form of a matrix A yields m invariant subspaces $V_1 \subset \ldots \subset V_m$, the parameter m is bounded from above by m_A. The number m_A equals the number of irreducible factors of the characteristic polynomial of A, counted with multiplicities. Thus

Corollary 2. *The expected value of m_A is $O(\log n)$.*

For clarity, the complexity of some algorithms will be given in terms of n and m_A. This will lead to average case complexities in terms of n and $\log n$. Although the algorithms here presented all are deterministic, the complexity is a random variable (for an average distribution of matrices) whose expectation is bounded from above.

We show some results about the complexity of some computations with a Shift-Hessenberg matrix. We also recall how some problems concerning companion matrices can be fast solved.

First observe that a Shift-Hessenberg is a sparse matrix, with at most $m+1$ non-zero entries in each row. This leads to the following lemma.

Lemma 6. *Let H be a Shift-Hessenberg matrix of size n, and let M be any matrix of size $n \times n'$. Then product HM can be computed at cost $O(mnn')$.*

Furthermore a Shift-Hessenberg matrix has some properties regarding cyclicity, as already seen in definition 4, which can be exploited for reducing costs.

Definition 7. *Let H in $M_n(\mathbf{k})$ be a Shift-Hessenberg matrix. A matrix A is* polycyclic *for H if its columns have the form:*

$$\left[v_1, Hv_1, \dots H^{n_1-1}v_1, v_2, Hv_2, \dots, H^{n_2-1}v_2, \dots, v_m, Hv_m, \dots, H^{n_m-1}v_m\right],$$

where $n_1, n_2, \dots n_m$ are the sizes of the diagonal blocks of H, and $v_1, v_2, \dots, v_m$ are vectors of $\mathbf{k}^n$.

Proposition 1. *Let H be a Shift-Hessenberg matrix, let A,B be two matrices which are polycyclic for H. Let α, β be any field elements, then $\alpha A + \beta B$, I_n, H, HA and HB all are polycyclic for H.*

In other words, polycyclic matrices for H form $\mathbf{k}[H]$-submodule of $M_n(\mathbf{k})$.

Proposition 2. *Let H be a Shift-Hessenberg matrix of parameter m. Then the product HA can be obtained with complexity $O(mn^2)$ for any matrix A in $M_n(\mathbf{k})$ and with complexity $O(m^2n)$ whenever A is polycyclic for H. A polynomial $p(X)$ of degree at most t can be evaluated at H with complexity $O(tm^2n)$.*

Proof. Let A be a polycyclic matrix for A. The product HA is performed by modifying A as follows. Delete v_1, shift all vectors to the left. Then replace v_2 by $HH^{n_1-1}v_1 \dots$, v_m by $HH^{n_{m-1}-1}v_{m-1}$. Finally, put $HH^{n_m-1}v_m$ as nth column. The whole cost is $m(mn)$.

Let p be a polynomial of degree less or equal than t. We apply Horner's rule for evaluating a polynomial $p(H) = p_tH^t + p_{t-1}H^{t-1} + \cdots + p_1H + p_0I$. We compute $h_1 = p_tH + p_{t-1}I$, $h_2 = Hh_1 + p_{t-2}I, \dots, h_t = Hh_{t-1} + t_0I$. Each matrix h_i is computed from h_{i-1} at a cost $O(m^2n)$, thus a total cost of $O(tm^2n)$ for $p(H)$. □

We recall very simple and efficient procedures for solving relations involving a companion matrix, which can be found in [5]. From now on, given a companion matrix C with minimal polynomial $\pi(X)$ of degree n, the vector $(v_0, \dots v_{n-1})$ is identified with the polynomial $v(X) = v_0 + v_1X + v_2X^2 + \cdots + v_{n-1}X^{n-1}$.

Lemma 7. *Let C be a companion matrix with minimal polynomial $\pi(X)$, let v in $\mathbf{k}^n$, let $P(X)$ be a polynomial of degree at most n. Then:*

1. *Cv is computed at cost $2n$.*
2. *$P(C)$ is computed at cost $O(n^2)$.*
3. *If $P(X)$ is prime to $\pi(X)$, the equation at u: $P(C)u = v$ is solved at cost $O(n^2)$.*
4. *The minimal polynomial $\pi_v(X)$ of C relatively to v is computed at cost $O(n^2)$.*

Using proposition 2, and computing a SHS form at each step of recursion, the algorithm described in section 2 can be modified to get the following result.

Theorem 8. *Given the factorization of its characteristic polynomial, the minimal polynomial of any matrix A and a block-diagonal matrix similar to A exhibiting its characteristic subspaces, is computed with time complexity $O(n^3 + m_A^2 n^2)$.*

4. A direct algorithm for the minimal polynomial

We now give another algorithm for computing the minimal polynomial of a matrix $\mathbf{A}$, given a Shift-Hessenberg form for $\mathbf{A}$. This algorithm is a direct algorithm, and it does not require any previous knowledge on the characteristic polynomial. The drawback is that is does not produce a diagonal-block decomposition of $\mathbf{k}^n$ into the characteristic subspaces of $\mathbf{A}$.

Assume that we are given a Shift-Hessenberg form $\mathbf{H}$ for matrix $\mathbf{A}$. Then $\mathbf{H}$ is described by blocks as follows:

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_{B_1,B_1} & \mathbf{H}_{B_1,B_2} & \cdots & \mathbf{H}_{B_1,B_m} \\ 0 & \mathbf{H}_{B_2,B_2} & \cdots & \mathbf{H}_{B_2,B_m} \\ \vdots & & \ddots & \vdots \\ 0 & \cdots & 0 & \mathbf{H}_{B_m,B_m} \end{bmatrix}.$$

Notation 2. *We denote by B_k the set of indices of block k, and $\mathbf{k}^{B_k}$ is the corresponding vector space. We also denote by $B_{\geq k}$ the set of indices $B_k \cup B_{k+1} \ldots \cup B_m$. For any matrix $\mathbf{A} \in M_n(\mathbf{k})$ we denote by $\mathbf{A}_{B_i,B_j}$ the matrix obtained from rows in B_i and columns in B_j. We denote by $\mathbf{A}_{B_{\geq k}}$ the square matrix obtained from rows and columns from the $\mathbf{k}^{\text{th}}$ block up to the end. For a Shift-Hessenberg matrix, the minimal polynomial of the companion matrix H_{B_i,B_i} is denoted by $f_i(X)$, $i = 1, \ldots, m$.*

4.1. Nested ideals related to H.

Let I_k denote the set of polynomials $g(X) \in \mathbf{k}[X]$ such that:

$$g(H)_{B_i,B_i} = 0, \quad i = 1, \ldots, m, \text{ and } g(H)_{B_i,B_j} = 0, \quad k \leq i < j \leq m.$$

It is in fact an ideal and the inclusions $I_1 \subseteq I_2 \cdots \subseteq I_m$ hold. Let $p_k(X)$ be the monic generator of I_k, then $p_k \mid p_{k-1}$. We denote by $\phi_k(X)$ the polynomial such that $p_k(X) = \phi_k(X)p_{k+1}(X)$. Consider the case where $k = 1$. For $g(X) \in I_1$, we have that $g(H)_{B_i,B_j} = 0$, $1 \leq i \leq j \leq m$ and thus $g(H) = 0$. The ideal $I_1 = (p_1(X))$ is the ideal annihilating the matrix H, and p_1 is the minimal polynomial of H.

Now let $g(X) \in I_{k+1}$, then, focusing on blocks with row index set B_k and column index set B_j, $k \leq j \leq m$, we consider the result of computations with H and we obtain the relation:

$$(Hg(H))_{B_k,B_j} = H_{B_k,B_k}(g(H))_{B_k,B_j}.$$

Let $p(X)$ be a polynomial of the form $q(X)p_{k+1}(X)$, which is the general form for polynomials in I_{k+1}. We have that:

$$p(H)_{B_k,B_j} = q(H)_{B_k,B_k}(p_{k+1}(H))_{B_k,B_j} = q(H_{B_k,B_k})(p_{k+1}(H))_{B_k,B_j}, k \leq j \leq n.$$

Then $p(H)_{B_k,B_j} = 0$, $k \leq j \leq n$ if and only if $q(H_{B_k,B_k})(p_{k+1}(H))_{B_k,B_j} = 0$, $k \leq j \leq n$, i.e. if and only if $q(H_{B_k,B_k})$ annihilates the space generated by columns of all matrices $(p_{k+1}(H))_{B_k,B_j}$, $j = k, \ldots, m$. Thus we have:

Lemma 8. *The polynomial $\phi_k(X)$ is the minimal polynomial of the restriction of H_{B_k,B_k} to the subspace generated by the columns of the matrices $p_{k+1}(H)_{B_k,B_j}$, $j = k \ldots m$.*

Notice that, since $f_k(X)$ is the minimal polynomial of H_{B_k,B_k}, $\phi_k(X)$ divides $f_k(X)$.

4.2. **The algorithm for the minimal polynomial of H.** The algorithm consists in constructing $p_m(X), p_{m-1}(X) \ldots p_1(X)$, step by step, by actually computing the polynomials $\phi_k(X)$, using Lemma 8.
First step: polynomial $p_m(X)$ is to be computed. Since all diagonal blocks of $p_m(\mathbf{H})$ vanish, then $p_m(X)$ is the least common multiple of the $f_i(X)$, $i = 1, \ldots, m$.
Iterative step: computing $\phi_k(X)$ from the data of $p_{k+1}(X)$. We have:

$$p_{k+1}(\mathbf{H}) = \begin{bmatrix} 0 & p_{k+1}(\mathbf{H})_{B_1,B_2} & \cdots & \cdots & p_{k+1}(\mathbf{H})_{B_1,B_m} \\ & \ddots & & & \\ & & 0 & p_{k+1}(\mathbf{H})_{B_k,B_{k+1}} & p_{k+1}(\mathbf{H})_{B_k,B_m} \\ & & 0 & 0 & 0 \\ & & 0 & 0 & 0 \end{bmatrix}.$$

From Lemma 8, $\phi_k(X)$ is the minimal polynomial of $\mathbf{H}_{B_k,B_k}$ restricted to the subspace generated by the columns of matrices $p_{k+1}(\mathbf{H})_{B_k,B_j}$, $k \leq j \leq m$.

Let ${}^t a^1 = (a^1_1, a^1_2, \ldots, a^1_{m_k})$ be the first non zero-column of the array formed by all matrices $p_{k+1}(\mathbf{H})_{B_k,B_j}$, $j \geq k$. Using Lemma 7, we compute the minimal polynomial $\phi_{k,a^1}(X)$ of $\mathbf{H}_{B_k,B_k}$ restricted to a^1. Thus $\phi_{k,a^1}(X)$ is a factor of $\phi_k(X)$ and we then compute $\mathbf{H}_{k,a^1} = \phi_{k,a^1}(\mathbf{H})p_{k+1}(\mathbf{H})$. The process is repeated on the first non-zero column a^2 of column of $\mathbf{H}_{k,a^1}$, to get a new factor $\phi_{k,a^2}(X)$ of $\phi_k(X)$. We compute again $\mathbf{H}_{k,a^2} = \phi_{k,a^2}(\mathbf{H})\mathbf{H}_{k,a^1}$, and proceed with the first

non-zero column of the array $\{(\mathbf{H}_{k,a^2})_{B_k,B_j}\}, j \geq k$. The process is stopped when all columns are canceled. We then have that $\phi_k(X) = \phi_{k,a^1}(X)\phi_{k,a^2}(X)\cdots\phi_{k,a^l}(X)$, where a^l is the last non-zero column which was met.

4.3. **Complexity bounds.** The most expensive computations lie in the computation of matrices $p_m(\mathbf{H})$, $p_{m-1}p_m(\mathbf{H})$, $p_{m-2}p_{m-1}p_m(\mathbf{H})\ldots$ The total cost is thus bounded by the cost of evaluating polynomials of degree less than n at a Shift-Hessenberg matrix, which is, by proposition 2, $O(m^2n^2)$ a number of times which is bounded by $m_{\mathbf{A}}$, the number of factors of the characteristic polynomials.

Each computation of a minimal polynomial ϕ_{k,a^i} is done at cost $O(n_k^2)$. The number of such computations is also bounded by $m_{\mathbf{A}}$. This results in $O(m_{\mathbf{A}}n^2)$ elementary operations for all those gcd computations.

Theorem 9. *The minimal polynomial of a Shift-Hessenberg matrix can be obtained in $O(m_{\mathbf{A}}m^2n^2)$ elementary operations. The minimal polynomial of any matrix $\mathbf{A}$ can be obtained in $O(n^3 + m_{\mathbf{A}}^3n^2)$ elementary operations without any previous knowledge on the characteristic polynomial.*

The term in n^3 is only due to computing a Shift-Hessenberg form of matrix A.

Remark 1. *Note that the worst case complexity, when m is n, is $O(n^5)$, which is bad. An alternative technique can be used for computing the matrices $p_m(\mathbf{H})$, $p_{m-1}p_m(\mathbf{H})\ldots$*

Let $d_1,\ldots,d_m$ be the degrees of the polynomials $p_1,p_2,\ldots,p_m$. First $p_m(\mathbf{H})$ is computed at cost d_mm^2n by proposition 2. Let $\mathbf{C}_{k+1} = p_{k+1}p_k\cdots p_m(\mathbf{H})$, which is a polycyclic matrix for $\mathbf{H}$, and let $p_k(X) = X^{d_k} + a_{d_k-1}X^{d_k-1} + \cdots + a_1X + a_0$. Compute $p_k(\mathbf{H})\mathbf{C}_{k+1}$ as follows:

$$\begin{aligned} p_k(\mathbf{H})\mathbf{C}_{k+1} &= (\mathbf{H}^{d_k} + a_{d_k-1}\mathbf{H}^{d_k-1} + \cdots + a_1\mathbf{H} + a_0)\mathbf{C}_{k+1} \\ &= (\mathbf{H}^{d_k-1} + a_{d_k-1}\mathbf{H}^{d_k-2} + \cdots + a_1)\mathbf{H}\mathbf{C}_{k+1} + a_0\mathbf{C}_{k+1}. \end{aligned}$$

Now the product $\mathbf{H}\mathbf{C}_{k+1}$ is computed at cost $O(m^2n)$ by Lemma 2, and the product $a_0\mathbf{C}_{k+1}$ at cost $O(n^2)$, and the sum of these two matrices is computed at cost $O(n^2)$. Thus computing $p_k(\mathbf{H})\mathbf{C}_{k+1}$ is performed at cost $O(d_k(m^2n+n^2))$, and the final cost is $O((d_1+\cdots+d_m)(m^2n+n^2)) = O(m^2n^2+n^3)$.

Using previous remark, we thus have:

Corollary 3. *The minimal polynomial of any matrix* $\mathbf{A}$ *can be obtained in* $O(n^3+m_{\mathbf{A}}^2n^2)$ *elementary operations without any previous knowledge on the characteristic polynomial.*

5. Searching for a cyclic vector

Let us recall some definitions.

Theorem 10. [4, Ch. VII §3 th. 2] *For all* $\mathbf{A}$ *in* $M_n(\mathbf{k})$*, there exists a vector* v *in* $\mathbf{k}^n$ *such that* $\pi_v(X) = \pi(X)$ *where* $\pi(X)$ *is the minimal polynomial of* $\mathbf{A}$*.*

Definition 8. *Let* $\mathbf{A}$ *be a matrix in* $M_n(\mathbf{k})$*. A vector* v *in* $\mathbf{k}^n$ *such that* $\pi_v(X) = \pi(X)$*, where* $\pi(X)$ *is the minimal polynomial of* $\mathbf{A}$*, is called a* cyclic vector *for* $\mathbf{A}$*.*

First we here show how to compute a cyclic vector at cost $O(m^3 + m^2n^2)$ for a square matrix $\mathbf{A}$ whose characteristic polynomial is square-free. This implies that the minimal polynomial of $\mathbf{A}$ equals its characteristic polynomial. Also the minimal polynomials $f_k(X)$ of the diagonal companion matrices of a Shift-Hessenberg form for $\mathbf{A}$ are pairwise relatively prime.

5.1. Technical lemmas.

The following lemma sets up the recurrence which ends in the sought for cyclic vector.

Notation 3. *Given a vector* v *in* $\mathbf{k}^n$*, the vector of size* n_I*, which is the projection of* v *into* $\mathbf{k}^{B_I}$*, is denoted by* v_{B_I}*. We denote by* $v^\star_{B_I}$ *the unique vector of* $\mathbf{k}^n$ *such that its projection into* $\mathbf{k}^{B_I}$ *equals* v_{B_I} *and such that its projection into* $\mathbf{k}^{B_J}$ *is 0, where* J *is the complementary set of* I *in* $[1, n]$*:* $(v^\star_{B_I})_{B_J} = 0$*.*

Lemma 9. *Let* $\mathbf{A}$ *be a block matrix with the form:*

$$\begin{bmatrix} \mathbf{A}_{B_1,B_1} & \mathbf{A}_{B_1,B_2} \\ 0 & \mathbf{A}_{B_2,B_2} \end{bmatrix},$$

and let v_{B_1}*,* v_{B_2} *be cyclic vectors for* $\mathbf{A}_{B_1,B_1}$ *and* $\mathbf{A}_{B_2,B_2}$ *respectively, matrices with respective minimal polynomials* $f_1(X)$ *and* $f_2(X)$*. If* $f_1(X)$ *and* $f_2(X)$ *are relatively prime, then the relations:*

$$v_{B_2} = u_{B_2}, \tag{4}$$

$$v_{B_1} = f_2(\mathbf{A}_{B_1,B_1})u_{B_1} + (f_2(\mathbf{A})u^\star_{B_2})_{B_1}, \tag{5}$$

can be solved at $u = (u_{B_1}, u_{B_2})$*, and the unique solution* u *is a cyclic vector for* $\mathbf{A}$*.*

Proof. The solution u is obtained by finding u_{B_1}. Since $f_1(X)$ and $f_2(X)$ are coprime, there exists $h_2(X)$ such that $f_2(X)h_2(X) = 1 \bmod f_1(X)$. Thus the matrix $h_2(A_{B_1,B_1})$ is the inverse of $f_2(A_{B_1,B_1})$, and the existence and unicity of u_{B_1} is guaranteed. Now we have to prove that $f_1(X)f_2(X)$ is the minimal polynomial of $u = (u_{B_1}, u_{B_2})$. Assume that $p(\mathbf{A})u = 0$ for a non-zero polynomial $p(X)$ with minimal degree. Then $p(X)$ is a divisor of $f_1(X)f_2(X)$ and we must have that $p(X) = p_1(X)p_2(X)$ with the condition that $p_1(X) \mid f_1(X)$, $p_2(X) \mid f_2(X)$ and $\gcd(p_1(X), p_2(X)) = 1$. The relation $p(\mathbf{A})u = 0$ gives:

$$p_1(\mathbf{A}_{B_2,B_2})p_2(\mathbf{A}_{B_2,B_2})u_{B_2} = 0. \tag{6}$$

Since $\gcd(p_1(X), f_2(X)) = 1$, there exists $h_1(X)$ such that $p_1(X)h_1(X) = 1 \bmod f_2(X)$. Applying $h_1(\mathbf{A}_{B_2,B_2})$ on both sides of (6) we get $p_2(\mathbf{A}_{B_2,B_2})u_{B_2} = 0$. Since u_{B_2} is a cyclic vector for $\mathbf{A}_{B_2,B_2}$, $f_2(X)$ divides $p_2(X)$ and $p_2(X) = f_2(X)$.

On the first block of coordinates, the equation $p(\mathbf{A})u = 0$ writes:

$$p_1(\mathbf{A}_{B_1,B_1})\left(f_2(\mathbf{A}_{B_1,B_1})u_{B_1} + (f_2(\mathbf{A})u^\star_{B_2})_{B_1}\right) = 0. \tag{7}$$

By hypothesis, $f_2(\mathbf{A}_{B_1,B_1})u_{B_1} + (f_2(\mathbf{A})u^\star_{B_2})_{B_1} = v_{B_1}$ is cyclic for $\mathbf{A}_{B_1,B_1}$. Then by (7), $f_1(X) \mid p_1(X)$ and $p_1(X) = f_1(X)$. □

We observe the striking fact that those computations can be performed at low cost.

Lemma 10. *A solution u to equations (4) and (5) may be computed in $O(n^3)$ elementary operations.*

Proof. First compute $w_{B_1} = (f_2(\mathbf{A})u^\star_{B_2})_{B_1}$, at cost $O(n^3)$. Then solve equation (5) by finding an inverse $h_2(X)$ of $f_2(X) \bmod f_1(X)$. Then the solution u_{B_1} is given by $u_{B_1} = h_2(\mathbf{A}_{B_1,B_1})(v_{B_1} - w_{B_1})$, calculated with complexity $O(n^3)$. □

5.2. **The naïve recurrence.** We denote by $u_{B_{\geq k}}$ *a cyclic vector for* $\mathbf{H}_{B_{\geq k}}$. The aim is to find $u_{B_{\geq 1}}$.
First step: The last block $\mathbf{H}_{B_m,B_m}$ is a companion matrix, the vector ${}^t(1, 0, \ldots, 0)$ is a cyclic vector for $\mathbf{H}_{B_m,B_m}$ and is choosed for u_{B_m}.
Iterative step: Suppose that the problem has been solved for $\mathbf{H}_{B_{\geq k+1}}$, i.e. we have a vector $u_{B_{\geq k+1}}$ which is cyclic for $\mathbf{H}_{B_{\geq k+1}}$. The minimal polynomial of $\mathbf{H}_{B_{\geq k+1}}$ is $f_{k+1}f_{k+2}\cdots f_m$, and the minimal polynomial of H_{B_k,B_k} is $f_k(X)$. These polynomials are coprime, and the Lemma 9 can be used to construct $u_{B_{\geq k}} = (u_{B_k}, u_{B_{\geq k+1}})$ which is cyclic for $\mathbf{H}_{B_{\geq k}}$.
End The result is $w_{B_{\geq 1}}$.

We now evaluate the number of operations to be performed to achieve the recurrence. The most expensive calculations lie in computing the vectors:

$$\begin{aligned} w_{B_{m-1}} &= (f_m(\mathbf{H})u^\star_{B_m})_{B_{m-1}}, \\ w_{B_{m-2}} &= ((f_{m-1}f_m)(\mathbf{H})u^\star_{B\geq m-1})_{B_{m-2}}, \\ &\vdots \\ w_{B_k} &= ((f_{k+1}f_{k+2}\cdots f_m)(\mathbf{H})u^\star_{B\geq k+1})_{B_k}. \end{aligned}$$

Computing each vector w_{B_k} consists mainly in applying at most n times matrix $\mathbf{H}$ at vectors with n components. The cost is $n \cdot mn$ for each of the m values of k. Moreover each u_{B_k} needs $O(n^2 m)$ steps and a separate cost of $O(n_k^2)$ is required for computing each of m gcd's. Taking into account the computation of $\mathbf{H}$ itself, this amounts to $O(m^2n^2 + n^3)$ elementary operations.

Theorem 11. *If the characteristic polynomial of the matrix* $\mathbf{A}$ *is square-free, a cyclic vector for* $\mathbf{A}$ *can be obtained in* $O(n^3 + m_{\mathbf{A}}^2 n^2)$ *steps.*

6. Obtaining a cyclic vector in $O(n^3)$ elementary operations

The previous procedure is not efficient for large m. We thus develop a more sophisticated procedure, whose complexity is $O(n^3)$, for any value of m.

The present algorithm computes a cyclic vector for a matrix whose minimal polynomial is square-free. The algorithm uses a "divide-and-conquer" approach as in Section 2. We first present its global structure, before going into details. We also set out separately a technique of splitting, and finally give the complete description.

6.1. Overall strategy.

First a Shift-Hessenberg form for the given matrix is to be computed. Then our strategy is to split it into two parts, whose sizes remain under control. The matrix $\mathbf{H}$ has the following form:

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_{B_1,B_1} & \mathbf{H}_{B_1,B_2} & \cdots & \mathbf{H}_{B_1,B_m} \\ 0 & \mathbf{H}_{B_2,B_2} & \cdots & \mathbf{H}_{B_2,B_m} \\ \vdots & & \ddots & \vdots \\ 0 & \cdots & 0 & \mathbf{H}_{B_m,B_m} \end{bmatrix},$$

Notation 4. *For every* $I \subset [1,n], J \subset [1,n]$, *we denote by* $\mathbf{H}_{I,J}$ *the sub-matrix formed with the rows of* $\mathbf{H}$ *in* I *and the columns of* $\mathbf{H}$ *in* J. *The size of* I *is denoted by* n_I. *Whenever* I *is reduced to a block* B_k *then the size of* I *is denoted by* n_k.

The splitting consists in finding a matrix $\mathbf{H}_{split}$ similar to $\mathbf{H}$ with the form:

$$\mathbf{H}_{split} = \begin{bmatrix} \mathbf{H}'_{B_I,B_I} & \mathbf{H}'_{B_I,B_J} \\ 0 & \mathbf{H}'_{B_J,B_J} \end{bmatrix}, \tag{8}$$

which moreover is a Shift-Hessenberg matrix, such that $n_I \leq \frac{2}{3}n, n_J \leq \frac{2}{3}n$. We recursively apply the algorithm on both matrices $\mathbf{H}'_{B_I,B_I}$ and $\mathbf{H}'_{B_J,B_J}$, in order to find v_{B_I}, v_{B_J} which are cyclic vectors of $\mathbf{H}'_{B_I,B_I}$ and $\mathbf{H}'_{B_J,B_J}$ respectively.

It remains to compute a vector u' cyclic for $\mathbf{H}_{split}$, v_{B_I} and v_{B_J} being known. Changing the current basis for the original one , we finally transform u' into a cyclic vector u for $\mathbf{H}$.

6.2. **The splitting.** We give a lemma for splitting the matrix into two submatrices. Before stating this lemma, we explain a technical but important phenomenom that appears when permuting rows and columns of Shift-Hessenberg matrices in order to move the blocks. Consider the following Shift-Hessenberg matrix:

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_{B_1,B_1} & \mathbf{H}_{B_1,B_2} & \cdots & \mathbf{H}_{B_1,B_k} & \cdots & \mathbf{H}_{B_1,B_m} \\ 0 & \mathbf{H}_{B_2,B_2} & \cdots & \mathbf{H}_{B_2,B_k} & \cdots & \mathbf{H}_{B_2,B_m} \\ \vdots & & \ddots & \vdots & \cdots & \vdots \\ \vdots & & & \mathbf{H}_{B_k,B_k} & \cdots & \vdots \\ 0 & 0 & \cdots & 0 & \cdots & \mathbf{H}_{B_m,B_m} \end{bmatrix}.$$

Let us perform the permutation of rows and columns which exchanges $\mathbf{H}_{B_1,B_1}$ $\mathbf{H}_{B_k,B_k}$. This leads to the matrix $\mathbf{H}_{swap}$:

$$\mathbf{H}_{swap} = \begin{bmatrix} \mathbf{H}_{B_k,B_k} & 0 \cdots 0 & 0 & \mathbf{H}_{B_k,B_{>k}} & \\ \mathbf{H}_{B_{[2,k-1]},B_k} & \mathbf{H}_{B_{[2,k-1]},B_{[2,k-1]}} & \mathbf{H}_{B_{[2,k-1]},B_1} & \mathbf{H}_{B_{[2,k-1]},B_{>k}} & \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{H}_{B_1,B_k} & \mathbf{H}_{B_1,B_{[2,k-1]}} & \mathbf{H}_{B_1,B_1} & \mathbf{H}_{B_1,B_{>k}} & \\ \mathbf{H}_{B_{>k},B_k} & \mathbf{H}_{B_{>k},B_{[2,k-1]}} & \mathbf{H}_{B_{>k},B_1} & \mathbf{H}_{B_{>k},B_{>k}} & \end{bmatrix}.$$

We now use the algorithm for computing a Shift-Hessenberg form of $\mathbf{H}_{swap}$. This leads to the matrix:

$$\mathbf{H}' = \begin{bmatrix} \mathbf{H}'_{B_1,B_1} & \mathbf{H}'_{B_1,B_2} & \cdots & \mathbf{H}'_{B_1,B_k} & \cdots & \mathbf{H}'_{B_1,B_m} \\ 0 & \mathbf{H}'_{B_2,B_2} & \cdots & \mathbf{H}'_{B_2,B_k} & \cdots & \mathbf{H}'_{B_2,B_m} \\ \vdots & & \ddots & \vdots & \cdots & \vdots \\ \vdots & & & \mathbf{H}'_{B_k,B_k} & \cdots & \vdots \\ 0 & 0 & \cdots & 0 & \cdots & \mathbf{H}'_{B_m,B_m} \end{bmatrix}.$$

The next lemma establishes a relation between the companion polynomial of block $\mathbf{H}'_{B_1,B_1}$ and the companion polynomial of block $\mathbf{H}_{B_k,B_k}$.

Notation 5. *We denote by ϵ_k the vector from the basis of k^n such that $(\epsilon_k)_{B_k} = (1, 0, \ldots, 0)$.*

Lemma 11. *Let f_k be the companion polynomial of block $\mathbf{H}_{B_k,B_k}$ of the matrix $\mathbf{H}$, and let f'_1 be the companion polynomial block $\mathbf{H}'_{B_1,B_1}$ of the matrix $\mathbf{H}'$ obtained in the previous transformations. We have that f_k divides f'_1.*

Proof: We have that f_k divides the minimal polynomial of $\mathbf{H}$ relatively to ϵ_k. Swapping from $\mathbf{H}$ to $\mathbf{H}_{swap}$ is placing vector ϵ_k as the first vector of the new basis.

The Shift-Hessenberg reduction algorithm computes a matrix whose first block is a companion matrix whose companion polynomial is the minimal polynomial of the first vector. Thus f'_1 is the minimal polynomial of ϵ_k, which is a multiple of f_k. □

Now we can state our important lemma for splitting Shift-Hessenberg matrices:

Lemma 12 (Splitting the matrix)**.** *Let $\mathbf{H}$ be a Shift-Hessenberg matrix. It is always possible to find a Shift-Hessenberg matrix $\mathbf{H}_{split}$ and an invertible matrix $\mathbf{P}$ such that $\mathbf{H} = \mathbf{P}\mathbf{H}_{split}\mathbf{P}^{-1}$ with $\mathbf{H}_{split}$ of the form:*

$$\mathbf{H}_{split} = \begin{bmatrix} \mathbf{H}'_{B_I,B_I} & \mathbf{H}'_{B_I,B_J} \\ 0 & \mathbf{H}'_{B_J,B_J} \end{bmatrix}, \tag{9}$$

and such that one of those three possibilities occurs:

1. *either $\mathbf{H}'_{B_I,B_I}$ is a companion block with size $\geq \frac{2}{3}n$, and $\mathbf{H}'_{B_J,B_J}$ has size $\leq \frac{1}{3}n$.*
2. *or $\mathbf{H}'_{B_I,B_I}$ is a companion block with size $\leq \frac{2}{3}n$, and $\mathbf{H}'_{B_J,B_J}$ has size $\leq \frac{2}{3}n$.*
3. *or both blocks $\mathbf{H}'_{B_I,B_I}$ and $\mathbf{H}'_{B_J,B_J}$ are nothing else that Shift-Hessenberg matrices with size smaller than $\frac{2}{3}n$.*

The computation of $\mathbf{H}_{split}$ and $\mathbf{P}$ can be performed in $O(n^3)$ operations.

Proof: Two main distinct cases are first considered.

First case: there exists $k \in [1, m]$ such that $n_k \geq \frac{2}{3}n$. Choose $I = B_k$, $J = [1, m] \setminus I$. We have that $n_J \leq \frac{1}{3}n$ but the block B_k may not be the first block. By permutations of rows and colums, block B_k is put in the first place. This gives a matrix $\mathbf{H}_{swap}$ which is not Shift-Hessenberg. We now can clean up matrix $\mathbf{H}_{swap}$ by applying the reduction algorithm producing a Shift-Hessenberg matrix. The size of the first block can

only grow, by Lemma 11, and then remains larger than $\frac{2}{3}n$. This gives matrix $\mathbf{H}_{split}$ shaped as in Case 1 at cost $O(n^3)$.

Second case: for each $j \in [1, m]$, $n_j < \frac{2}{3}n$. Suppose first that all n_i are smaller than $\frac{1}{3}n$. In the sequence of sets $I_i = \{1, 2, \dots i\}$, we choose the largest, I_{i_0} with the condition that $\sum_{j \in I_i} n_j < \frac{2}{3}n$. Then $I = B_1 \cup B_2 \cdots B_{i_0}$ and $J = B_{i_0+1} \cup B_{i_0+2} \cdots \cup B_m$ both satisfy $n_I \leq \frac{2}{3}n$ and $n_J \leq \frac{2}{3}n$. Indeed, since $n_{J \setminus B_{i_0+1}} < \frac{1}{3}n$, we have that $n_J < \frac{1}{3}n + n_{i_0+1} \leq \frac{2}{3}n$. Then the matrix $\mathbf{H}_{split}$ is the unchanged matrix $\mathbf{H}$. This is case 3.

If there exists $n_k \geq \frac{1}{3}n$, we choose $I = B_k$, $J = [1, m] \setminus I$. We have $n_I \leq \frac{2}{3}n$, $n_J \leq \frac{2}{3}n$. By swapping rows and columnns, we put the block $\mathbf{H}_{I,I}$ in the first place, then clean up the resulting matrix by the Shift-Hessenberg reduction algorithm in $O(n^3)$ steps. The first block can only grow. As a result the size of the remaining block remains lower than $\frac{2}{3}n$; if the size of the first block is larger than $\frac{2}{3}n$, then we are in Case 1, else we are in Case 2. □

6.3. **The algorithm itself.** We now present the complete algorithm for computing a cyclic vector for a matrix $\mathbf{A}$ such that its minimal polynomial is square-free.

Step 1*: computation of a Shift-Hessenberg form of $\mathbf{A}$. As stated in Theorem 7, this is done in $O(n^3)$ operations. This step needs only to be performed once, and is not needed in the recursive steps.

Step 2 : splitting the matrix. We perform the splitting indicated by Lemma 12, and obtain two submatrices $\mathbf{H'}_{B_I,B_I}$ and $\mathbf{H'}_{B_J,B_J}$.

We recursively apply the algorithm on all submatrices which occur with size $\leq \frac{2}{3}n$.

Step 3: reconstruction of a cyclic element in a new basis. We get the two vectors u_{B_1} and u_{B_2} for the equations (4) and (5) from the results of the algorithm applied at $\mathbf{H'}_{B_I,B_I}$ and $\mathbf{H'}_{B_J,B_J}$. By Lemma 10 we can construct a cyclic element for $\mathbf{H}_{split}$, at cost $O(n^3)$.

Step 4: reconstruction of the cyclic element in the original basis. From a cyclic vector of $\mathbf{H}_{split}$, changing basis gives a cyclic vector for $\mathbf{H}$ at cost $O(n^3)$.

Step 5*: reverting to the original basis. From a cyclic vector for $\mathbf{H}$, we compute a cyclic vector for $\mathbf{A}$ by changing basis. This costs $O(n^3)$, and is performed only once, at the end of the algorithm.

Theorem 12. *Given a matrix $\mathbf{A} \in M_n(k)$ whose minimal polynomial is square-free, a cyclic vector for $\mathbf{A}$ can be computed in $O(n^3)$ elementary operations.*

Proof. The proof is easily done by recurrence, as in 3, by observing that the cost of all intermediae steps before recursion are bound by $O(n^3)$. □

6.4. **Applications to normal bases.** Before stating our result, we recall that the best known complexity for deterministic algorithms for finding a normal basis is $O((n^2 + \log q)n^2)$ [2]. Considering probabilistic algorithms, J. Von zur Gathen [9] presents an algorithm with expected time $O^{\sim}(n^2 \log q)$, where $g = O^{\sim}(h)$ means that there exists k such that $g = O(h \log(h)^k)$ ("soft-O" notation). Fast algorithm for polynomial multiplication and for gcd's are used. When using classical arithmetic, the complexity of this algorithm turns to $O(n^3 + n^2 \log q)$ or $O(n^3 \log(n))$ depending on the relative size of q and n [5].

Corollary 4. *When n is prime to p, it is possible to compute a normal basis of $\mathbb{F}_{q^n}$ in $O(n^3)$ elementary operations on the data of a matrix representing the Frobenius map, that matrix being computed at cost $O(n^3 + n^2 \log q)$.*

Proof: The minimal polynomial of the Frobenius map is $X^n - 1$, which is square-free when $\gcd(n, q) = 1$. Given the matrix $\mathbf{F}_n$ of the Frobenius map (computed in $O(n^3 + n^2 \log q)$), we are able to compute a cyclic vector for the Frobenius map in $O(n^3)$. This vector is a normal element. □

We now consider the case where $n = p^t$, where p is the characteristic of the field. In that case, $X^n - 1 = (X - 1)^n$. Let H be a Shift-Hessenberg matrix for the Frobenius automorphism, and let $\epsilon_1, \ldots, \epsilon_m$ be basis vector as in notation 5. Since the minimal polynomial of H is $X^n - 1$, and is also the least common multiple of the minimum polynomial of the ϵ_i's, then $X^n - 1$ is the minimal polynomial of one of the ϵ_i. Now if ϵ_l were cyclic with $l < m$, we would permute the basis vectors in order to have ϵ_l in the first position. After reduction to Shift-Hessenberg form, it is seen that the last rows and columns would remain unchanged and in particular the zero in the subdiagonal located in the column preceding ϵ_m would remain unchanged. This contradicts the fact that ϵ_l is cyclic, since putting it in the first position would lead to a companion matrix.

To sum up, a reduction of any representation of the Frobenius map into a Shift-Hessenberg form exhibits ϵ_m which necessarily is cyclic. Knowing normal bases for $\mathbb{F}_{q^{n_1}}$ and $\mathbb{F}_{q^{n_2}}$, one can construct a normal element for $\mathbb{F}_{q^{n_1 n_2}}$, when $\gcd(n_1, n_2) = 1$ [1, 2].

Corollary 5. *For all n, a normal basis of $\mathbb{F}_{q^n}$ can be computed deterministically in $O(n^3 + n^2 \log q)$ elementary operations.*

7. Computation of the Frobenius Form

7.1. **Definitions and Notations.** Furthermore we need a specific notation for the columns of a Shift-Hessenberg matrix.

Notation 6. *Let* $\mathbf{H}$ *be a Shift-Hessenberg matrix:*

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_{B_1,B_1} & \mathbf{H}_{B_1,B_2} & \cdots & \mathbf{H}_{B_1,B_m} \\ 0 & \mathbf{H}_{B_2,B_2} & \cdots & \mathbf{H}_{B_2,B_m} \\ \vdots & & \ddots & \vdots \\ 0 & \cdots & 0 & \mathbf{H}_{B_m,B_m} \end{bmatrix}. \tag{10}$$

We recall that ϵ_i *the unit vector from* $\mathbf{k}^n$ *such that* $(\epsilon_i)_{B_i} = {}^t(1,\ldots,0)$ *(Notation 5). We set* $e_i = f_i\epsilon_i$*. Informally,* e_i *is seen to be the vector "above" block* i *in* H*.*

We here shall describe how to compute the Expanded-Frobenius form of a matrix. A preliminary computation is done, to compute a basis for the characteristic subspaces of A, using the algorithm presented in section 2 (and thus the factorization of the characteristic polynomial of A is required). Then the expanded frobenius form for each characteristic subspace is computed, with the following method.

7.2. **Computing the Frobenius form for characteristic subspaces.** We considerthe case where the characteristic polynomial of the considered matrix is $C(X) = p(X)^r$, with $r \geq 1$, and $p(X)$ irreducible.

We apply the reduction process to get a Shift-Hessenberg form $\mathbf{H}$ for the matrix, as in equation (10). The minimal polynomial of block H_{B_i,B_i} is thus p^{s_i} for some s_i.

The vector $(e_i)_{B_1}$ seen as a polynomial, powers of p can be extracted out of e_i, using gcd computations. We thus have $(e_i)_{B_1} = e_i^\dagger p^{r_i}$, with $e_i^\dagger$ prime to p. A favorable case occurs when each e_i is such that $r_i \geq s_i$. In that case, we introduce the vectors:

$$\epsilon_2' = \epsilon_2 - e_2^\dagger p^{r_2-s_2}\epsilon_1, \ldots, \epsilon_m' = \epsilon_m - e_m^\dagger p^{r_m-s_m}\epsilon_1.$$

For those vectors we have that $(p^{s_i}\epsilon_i')_{B_1} = 0$ since $p^{s_i}\epsilon_i = e_i$. The first vector ϵ_1 remains unchanged. The vectors $\epsilon_1, \epsilon_2' \ldots \epsilon_m'$ yield a basis in which the matrix has the following form:

$$\begin{bmatrix} \mathbf{C}_{p^{s_1}} & 0 \\ 0 & \mathbf{H}' \end{bmatrix}.$$

The process is next applied to $\mathbf{H}'$.

Otherwise, there exists ϵ_j such that $r_j < s_j$. We observe the following property:

Lemma 13. *Let ϵ_j be a vector such that $r_j < s_j$ in the above situation. Then the minimal polynomial of ϵ_j is p^t with $t > s_1$.*

Proof. To find the minimal polynomial of ϵ_j we have to find the lowest t such that $p^t\epsilon_j = 0$. We first have to compute $p^{s_j}\epsilon_j$ in order to have that $(p^{s_j}\epsilon_j)_{B_j}$ vanishes. This leads to $p^{s_j}\epsilon_j = e_j$. The coordinates of e_j on block B_{j-1} must vanish in their turn, so we apply the minimal polynomial of $(e_{j-1})_{B_{j-1}}$, which is $p^{\lambda_{j-1}}$ for some λ_{j-1}. We get $p^{\lambda_{j-1}}p^{s_j}\epsilon_j = p^{\lambda_{j-1}}(e_j)^\star_{B_1} + v(j-2)$, where $v(j-2)$ is a vector with support in blocks $B_1 \cup \cdots \cup B_{j-2}$. We proceed in this way and at each step we get a new relation $p^{\lambda_k+\cdots+\lambda_{j-1}+s_j}\epsilon_j = p^{\lambda_k+\cdots+\lambda_{j-1}}(e_j)_{B_1} + v(k-1)$, where $v(k-1)$ lies in $B_1 \cup \cdots \cup B_{k-1}$.

This ends after all other coordinates vanished except those in the first block. We then have that:

$$p^{\lambda_2+\cdots+\lambda_{j-1}+s_j}\epsilon_j = p^{\lambda_2+\cdots+\lambda_{j-1}}(e_j)_{B_1} + v(1) = p^{\lambda_2+\cdots+\lambda_{j-1}}e_j^\dagger p^{r_j} + v(1).$$

We thus are left with determining the minimum exponent l such that:

$$p^l\left(p^{\lambda_2}\cdots p^{\lambda_{j-1}}e_j^\dagger p^{r_j} + v(1)\right) = 0 \bmod p^{s_1},$$

and we write $v(1) = p^{r_0}v(1)^\dagger$ where $\gcd(p, v(1)^\dagger) = 1$. Two cases are to be considered.

If $r_0 \geq \lambda_2 + \cdots + \lambda_{j-1} + r_j$, then $l = s_1 - (\lambda_2 + \cdots + \lambda_{j-1} + r_j)$ and the exponent of the minimal polynomial of ϵ_j is:

$$t = l + \lambda_2 + \cdots + \lambda_{j-1} + s_j = s_1 - r_j + s_j > s_1.$$

In the other case, $r_0 < \lambda_2 + \cdots + \lambda_{j-1} + r_j$. Then $l = s_1 - r_0$, and the exponent of the minimal polynomial of ϵ_j is:

$$t = s_1 - r_0 + \lambda_2 + \cdots + \lambda_{j-1} + s_j > s_1 - r_j + s_j > s_1.$$

□

For completing the algorithm in that case, we permute the basis vectors in order to have ϵ_j in the first position. By applying the reduction algorithm, we compute a new Shift-Hessenberg form, whose first block is a companion matrix, with companion polynomial the minimal polynomial of ϵ_j. By previous lemma, the size of the first block has grown and as a result the sizes of the other blocks had to decrease. The process stops when we have $s_i \leq r_i$ for all i, and we apply the above method for the favourable case, or when we get a companion matrix.

7.3. **Complexity.** Either cleaning up the matrix when it is possible, or augmenting the size of the first block is done at cost $O(n^3)$. The number of times those processes are performed is bounded by r. Notice that matrices for changing bases are also obtained. Thus the complexity in the case of a characteristic subspace is bounded by $O(n^3 r)$. The complexity for all characteristic subspaces is bounded by:

$$O(n_1^3 r_1) + O(n_2^3 r_2) + \cdots + O(n_d^3 r_d) \leq O(n^3(r_1 + r_2 + \cdots + r_d)).$$

The number $r_1 + r_2 + \cdots + r_d$ is the number of factors of the characteristic polynomial counted with multiplicities. This number is $\log n$ on the average.

Theorem 13. *Knowing the factorization of its characteristic polynomial, the Frobenius form of a matrix* $\mathbf{A}$ *and the matrix for changing basis can be computed in* $O(n^3 m_{\mathbf{A}})$*, where* $m_{\mathbf{A}}$ *is the number of factors of the characteristic polynomial of* $\mathbf{A}$*, counted with multiplicities. The asymptotic average complexity over a finite field is* $O(n^3 \log n)$*.*

7.4. **Without the factorization of the characteristic polynomial.** We show how to perform the computation of the Frobenius form without the knowledge of the factorization of the characteristic polynomial. The idea is the following: the computation of the Shift-Hessenberg form of the matrix A yields a partial factorization of the characteristic polynomial$C$ of A. Using a "factor refinement" process, the characteristic polynomial C can be factorized into $C = P_1^{r_1} \dots P_k^{r_k}$, with $\gcd(P_i, P_j) = 1$, when $i \neq j$. The algorithm for theorem 8 can be applied to compute the restriction of the matrix to the subspaces $\ker P_i(A)^{r_i}$, $i = 1 \dots k$. Then the previous algorithm can be applied, making the (eventually false) assumption that the P_i's are irreducible. If some problem is encountered, then a newer factorization is obtained, and new subspaces are computed.

Here are the details of the algorithm:
Input Matrix A, (whose characteristic polynomial is denoted C).
Step 1 Computation of a Shift-Hessenbergform H for A. If a companion matrix is obtained, then the algorithm stops returning H.
Step 2 "Factor Refinement". The diagonal companion blocks of matrix H yield factors $f_1, \dots, f_m$, such that $f_1 \dots f_m = C$. The factor refinement is to extract pairwise gcd's from that list recursively, until we get a list $P_1^{r_1} \dots P_k^{r_k} = C$, with $\gcd(P_i, P_j) = 1$, for $i \neq j$. The subspaces $V_i = \ker P_i(A)^{r_i}$ are called pseudo-characteristic subpaces.
Step 3 Computing the restriction of H the pseudo-characteristic subpaces. This is done using the algorithm from theorem 8. This algorithm

is recursive, and at each step a Shift-Hessenberg form for each V_i is computed. Diagonal blocks appearing in Shift-Hessenberg forms along this process may show new factors P'_i, in which case the refinement process is applied, and new pseudo-characteristic subpaces are computed. This ends in the knowledge of the Shift-Hessenberg form of the restriction of H to new pseudo-characteristic subpaces $V'_i = \ker {P'}_i(A)^{r'_i}$, with $C = {P'}_1^{r'_1} \dots {P'}_{k'}^{r'_{k'}}$.

Step 4 Computing the Frobenius form for each pseudo-characteristic subspace. We apply the algorithm described in previous section, making the (possibly wrong) hypothesis that we are faced with a matrix H whose characteristic polynomial is p^r with p irreducible.

As previously, the vectors $(e_i)_{B_1}$ seen as polynomials, powers of p are extracted out of e_i, using repeated gcd computations. If all gcd's are powers of p, then we end with $(e_i)_{B_1} = e_i^\dagger p^{r_i}$, with $e_i^\dagger$ prime to p. If some gcd is not a power of p, then a factor of p has been found, and refinement for new subspaces is done, goto to **Step 3**. If it is not the case, then the same favorable case may appears as previously, and the process goes on (cleaning the first block). If the unfavorable case happens, then we put in first position a vector ϵ_i such that $r_i > s_i$, and apply the Shift-Hessenberg reduction. Then the minimal polynomial of ϵ_i appears on the first block: if it a power of p, then the size of the first block has grown (same argument as lemma 13, if not, a new factor of p appears, and refinement is done, goto to **Step 3**.

8. Conclusion

The efficiency of the presented algorithms is due to two major procedures here introduced.

The first one is the use of a divide-and-conquer algorithm which splits matrices of size n into sub-matrices of size $\leq \frac{2}{3}n$. Therefore we make the following remark: the cost of such an algorithm is the same as the cost for "dividing" and for "recombining" only once. The second is the use of the Shift-Hessenberg form, which is very sparse on the average, and which reflects some algebraic properties of the matrix. It can be computed at low cost and above all it allows one to make the most of the isomorphism from the algebra generated by the given matrix onto an algebra of polynomials by converting operations on matrices into operations on polynomials.

Considering the results of this paper, a natural question raises. Does there exist a deterministic algorithm for obtaining the Frobenius form of any matrix in $O(n^3)$ elementary operations on the average?

Acknowledgments

Daniel Lazard read the first draft of this paper and among very constructive criticisms drew our attention on the important results of Patrick Ozello. His remarks were encouragements to carry on with this venture. We also had a nice opportunity to get informed of the background of the present topic by Joachim von zur Gathen. We had an encouraging conversation with Arnold Schönhage and Jeremy Johnson informed us of a result by Richard Stong which was here generalized for the need of evaluating complexities.

References

[1] A. A. Albert. *Fundamental Concepts of Higher Algebra*. The University of Chicago Press, 1956.
[2] I.F. Blake, X.H. Gao, R.C. Mullin, S.A.Vanstone, and T. Yaghoobian. *Applications of finite fields*. Kluwer Academic Publishers, 1993.
[3] Henri Cohen. *A Course in Computational Algebraic Number Theory*. Springer-Verlag, 1993.
[4] F. R. Gantmacher. *The Theory of Matrices*, volume 1. Chelsea, 1977.
[5] Mark William Giesbrecht. *Nearly Optimal Algorithms for Canonical Matrix Forms*. PhD thesis, University of Toronto, 1993.
[6] Patrick Ozello. *Calcul exact des formes de Jordan et de Frobenius d'une matrice*. PhD thesis, Université Scientifique Technologique et Médicale de Grenoble, 1987.
[7] M.S. Paterson and L. Stockmeyer. On the number of nonscalar multiplications necessary to evaluate polynomials. *SIAM J. of Comput.*, 2:60–66, 1973.
[8] Richard Stong. Some asymptotic results on finite vector spaces. *Advances in Applied Mathematics*, 9:167–199, 1988.
[9] J. von Zur Gathen and M. Giesbrecht. Constructing normal bases in finite fields. *Journal of Symbolic Computation*, 10:547–570, 1990.
[10] J. von Zur Gathen and V. Shoup. Computing frobenius maps and factoring polynomials. *Computational Complexity*, 2:197–224, 1992.
[11] J. H. Wilkinson. *The Algebraic Eigenvalue Problem*. Oxford Science Publications, 1992.

Daniel Augot, Paul Camion, Projet Codes, INRIA Rocquencourt, Domaine de Voluceau, BP105, 78153, Le Chesnay Cedex
E-mail address: Daniel.Augot@inria.fr, Paul.Camion@inria.fr