Characteristic Polynomials of p-adic Matrices

Xavier Caruso
Université Rennes 1; IRMAR
Rennes, France 35042
xavier.caruso@normalesup.org

David Roe
University of Pittsburg; Department of Mathematics
Pittsburgh, PA, USA 15260
roed@pitt.edu

Tristan Vaccon
Université de Limoges; CNRS, XLIM UMR 7252
Limoges, France 87060
tristan.vaccon@unilim.fr

ABSTRACT

We analyze the precision of the characteristic polynomial χ_M of an $n \times n$ p-adic matrix M using differential precision methods developed previously. When M is an integral matrix whose entries are all given at the same precision $O(p^N)$, we give a criterion (checkable within $\tilde{O}(n^\omega)$ operations in $\mathbb{F}_p$) for the existence of a coefficient of χ_M with more accuracy than $O(p^N)$. In general, we provide two algorithms for determining the optimal precision of the coefficients of χ_M and of M's eigenvalues. We provide evidence showing that classical algorithms do not reach this optimal precision in general.

CCS CONCEPTS

• **Computing methodologies → Algebraic algorithms**;

KEYWORDS

Algorithms, p-adic precision, characteristic polynomial, eigenvalue

ACM Reference format:
Xavier Caruso, David Roe, and Tristan Vaccon. 2017. Characteristic Polynomials of p-adic Matrices. In *Proceedings of ISSAC '17, Kaiserslautern, Germany, July 25-28, 2017,* 8 pages.
https://doi.org/10.1145/3087604.3087618

1 INTRODUCTION

The characteristic polynomial is a fundamental invariant of a matrix: its roots give the eigenvalues, and the trace and determinant can be extracted from its coefficients. In fact, the best known division-free algorithm for computing determinants over arbitrary rings [11] does so using the characteristic polynomial. Over p-adic fields, computing the characteristic polynomial is a key ingredient in algorithms for counting points of varieties over finite fields (see [7, 8, 12]).

When computing with p-adic matrices, entries may only be approximated at some finite precision $O(p^N)$. As a consequence, in designing algorithms for such matrices one must analyze not only the running time of the algorithm but also the accuracy of the result.

Let $M \in M_n(\mathbb{Q}_p)$, with every entry given at precision $O(p^N)$. The simplest approach for computing the characteristic polynomial of M is to compute $\det(X-M)$ either using recursive row expansion or various division free algorithms [10, 16]. There are two issues with these methods. First, they are slower than alternatives that allow division, requiring $O((n+1)!)$, $\tilde{O}(n^5)$ and $\tilde{O}(n^{3+\omega/2})$ operations in K. Second, while the lack of division implies that the result is accurate modulo p^N as long as $M \in M_n(\mathbb{Z}_p)$, they still do not yield the optimal precision.

A faster approach over a field is to compute the Frobenius normal form of M, which is achievable in running time $\tilde{O}(n^\omega)$ [13, 15]. However, the use of division frequently leads to catastrophic losses of precision. In many examples, no precision remains at the end of the calculation.

Instead, we separate the computation of the precision of χ_M from the computation of an approximation to χ_M. Given some precision on M, we use the theory developed in [3] to find the best possible precision for χ_M. The analysis of this precision is the subject of much of this paper. With this precision known, the actual calculation of χ_M may proceed by lifting M to a temporarily higher precision and then using a sufficiently stable algorithm (see Remark 5.3).

Previous contributions. Since the description of Kedlaya's algorithm in [12], the computation of characteristic polynomials over p-adic numbers has become a crucial ingredient in many counting-points algorithms. For example, [7, 8] use p-adic cohomology and the characteristic polynomial of Frobenius to compute zeta functions of hyperelliptic curves.

In most of these papers, the precision analysis usually dwells on how to obtain the matrices (*e.g.* of action of Frobenius) that are involved in the point-counting schemes. However, the computation of their characteristic polynomials attracts less attention: some refer to fast algorithms (using division), while others apply division-free algorithms.

In [4], the authors have begun the application of the theory of differential precision of [3] to the stable computation of characteristic polynomials. They describe the optimal precision for the characteristic polynomial, but do not give practical algorithms to compute this optimal precision.

The contribution of this paper. This paper provides a theoretical and concrete study of the propagation of the precision during the computation of the characteristic polynomial of matrix defined over the p-adics. Thanks to the application of the general framework of differential precision developed in [3, 4], we know that the optimal precision of the characteristic polynomial χ_M of a matrix $M \in M_n(\mathbb{Q}_p)$ is controled by the adjugate $\operatorname{Adj}(X-M)$. In this article, we provide:

ISSAC '17, July 25-28, 2017, Kaiserslautern, Germany

ACM ISBN 978-1-4503-5064-8/17/07…$15.00
https://doi.org/10.1145/3087604.3087618

(1) Proposition 2.7: a short representation of the adjugate matrix $\mathrm{Adj}(X-M)$ whose size is $O(n^2)$ elements of the base field (instead of $O(n^3)$ for a dense representation);

(2) two algorithms to compute $\mathrm{Adj}(X-M)$: the first one is division-free and performs $\tilde{O}(n^3)$ operations in $\mathbb{Z}_p$ while the second one may perform divisions and has complexity $\tilde{O}(n^\omega)$;

and deduce from this the following applications to the question we are interested in:

(3) Corollary 2.4: a simple criterion to decide whether χ_M is defined at precision higher than the precision of M (when M lies in $M_n(\mathbb{Z}_p)$ and all its entries are given at the same precision);

(4) two algorithms to compute the optimal precision on each coefficient of the characteristic polynomial of a matrix $M \in M_n(\mathbb{Q}_p)$ (whose entries may be given at different precision): the first one is division-free and has complexity $\tilde{O}(n^3)$ (operations in $\mathbb{Z}_p$) while the second one may perform divisions but has complexity $\tilde{O}(n^\omega)$;

(5) Proposition 5.5: a $\tilde{O}(n^\omega)$ algorithm to compute the optimal precision on each eigenvalue of M.

Moreover, we provide evidence showing that classical algorithms for computing the characteristic polynomials do not reach the optimal precision in many contexts. Note that we focus on computing the precision of χ_M: we do not give a stable algorithm for computing the characteristic polynomial itself.

Organization of the article. In Section 2, we review the differential theory of precision developed in [3] and apply it to the specific case of the characteristic polynomial. We also describe when the characteristic polynomial will have a higher precision than the input matrix and give a compact description of $\mathrm{Adj}(X-M)$, the main ingredient in the differential.

In Section 3, we develop $\tilde{O}(n^3)$ algorithms to approximate the Hessenberg form of M, and through it to find $\mathrm{Adj}(X-M)$ and thus find the precision of the characteristic polynomial of M. In Section 4, we give a $\tilde{O}(n^\omega)$ algorithm to compute the compact description of $\mathrm{Adj}(X-M)$.

Finally, we propose in Section 5 algorithms to compute the optimal coefficient-wise precision for the characteristic polynomial. We also give the results of some experiments demonstrating that these methods can lead to dramatic gains in precision over standard interval arithmetic. We close by describing the precision associated to eigenvalues of a matrix.

Notations. Throughout the paper, K will refer to a complete, discrete valuation field, $\mathrm{val} : K \twoheadrightarrow \mathbb{Z} \cup \{+\infty\}$ to its valuation, $\mathcal{O}_K$ its ring of integers and π a uniformizer. We will write that $f(n) = \tilde{O}(g(n))$ if there exists some $k \in \mathbb{N}$ such that $f(n) = O(g(n)\log(n)^k)$. We will write M for an $n \times n$ matrix over K, and χ the characteristic polynomial map, $\chi_M \in K[X]$ for the characteristic polynomial of M and $d\chi_M$ for the differential of χ at M, as a linear map from $M_n(K)$ to the space of polynomials of degree less than n. We fix an $\omega \in \mathbb{R}$ such that the multiplication of two matrices over a ring is in $O(n^\omega)$ operations in the ring. Currently, the smallest known ω is less than 2.3728639 thanks to [14]. We will denote by I_n the identity matrix of rank n in $M_n(K)$. When there is no ambiguity, we will drop this I_n for scalar matrices, *e.g.* for $\lambda \in K$ and $M \in M_n(K)$, $\lambda - M$ denotes $\lambda I_n - M$. We write $\sigma_1(M), \dots, \sigma_n(M)$ for the elementary divisors of M, sorted in increasing order of valuation. Finally, given a polynomial $\chi(X) = a_0 + a_1 X + \cdots + a_{n-1}X^{n-1} + X^n$, the *companion matrix* $\mathscr{C}$ associated to χ is

$$\mathscr{C} = \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ 0 & 0 & \cdots & 0 & 1 \\ -a_0 & -a_1 & \cdots & \cdots & -a_{n-1} \end{pmatrix}. \tag{1}$$

2 THEORETICAL STUDY

2.1 The theory of p-adic precision

We recall some of the definitions and results of [3] as a foundation for our discussion of the precision for the characteristic polynomial of a matrix. We will be concerned with two spaces in what follows: the space $M_n(K)$ of $n \times n$ matrices with entries in K and the space $K_n[X]$ of monic degree n polynomials over K. We also define $K_{<n}[X]$ (resp. $\mathcal{O}_{K<n}[X]$) as the space of polynomials over K (resp. over $\mathcal{O}_K$) of degree strictly less than n.

One cannot represent matrices $M \in M_n(K)$ exactly, but instead must specify an approximation together with a measure of the approximation's accuracy; there are several ways to do so. When using *jagged* precision, one gives a precision on each entry, representing the (i, j) entry as $a_{i,j} + O(\pi^{N_{i,j}})$. The integer $N_{i,j}$ is called the *absolute precision* and may depend on (i, j). When the $N_{i,j}$ are all equal, we say that the precision datum is *flat*.

We may also model precision more flexibly by using lattices. We recall that a *lattice* H in $M_n(K)$ is the $\mathcal{O}_K$-span of a vector space basis, or equivalently the image of $M_n(\mathcal{O}_K)$ under a K-linear transformation of $M_n(K)$ [3, §2.2]. For $M \in M_n(K)$, cosets of the form $M + H$ are special neighborhoods of M; we will use them for modeling precision on M. Given a family of integers $\mathcal{N} = (N_{i,j})_{i,j}$ the lattice

$$H_{\mathcal{N}} = \left\{ M = (a_{i,j})_{i,j} \in M_n(\mathbb{Q}_p) \mid \forall i,j\colon \pi^{N_{i,j}} \text{ divides } a_{i,j} \right\}$$

corresponds to the jagged precision introduced above. The benefit of working with lattices is that they behave well under differentiable maps [3, Lemma 3.4], but they require more space and time to compute.

Jagged, flat and lattice precision is defined similarly for monic polynomials, taking care that no precision is specified for the leading coefficient which must be exactly 1. The underlying vector space is now $K_{<n}[X]$ and we define a lattice in it as the image of $\mathcal{O}_{K<n}[X]$ under some linear mapping.

Let $\chi : M_n(K) \to K_n[X]$ be the characteristic polynomial map and $d\chi_M$ for its differential at M. Our analysis of the precision behavior of χ rests upon the computation of its derivative $d\chi$, using [3, Lemma 3.4] Let $\mathrm{Adj}(M)$ denote the adjugate[1] of M (when $M \in \mathrm{GL}_n(K)$, we have $\mathrm{Adj}(M) = \det(M)M^{-1}$). Recall from [3, Appendix B] and [4, §3.3] that $d\chi_M$ is given by

$$\begin{array}{rccl} d\chi_M : & M_n(K) & \to & K_{<n}[X] \\ & dM & \mapsto & \mathrm{Tr}(\mathrm{Adj}(X-M) \cdot dM). \end{array} \tag{2}$$

[1] In [3, 4] the adjugate was referred to as the *comatrix*

When $d\chi_M$ is surjective and H is sufficiently "small" and "regular," we may give the precision of χ_M precisely as

$$\chi(M + H) = \chi(M) + d\chi_M(H). \tag{3}$$

The equality in (3) justifies our use of the phrase *optimal* precision to denote the lattice precision $d\chi_M(H)$. To make the conditions on H under which (3) holds more precise, let $\sigma_1(M), \ldots, \sigma_n(M)$ denote the elementary divisors of M.

Proposition 2.1. *Let $M \in M_n(K)$ and assume that $d\chi_M$ is surjective. Let:*

$$v = \max\left(\sum_{i=1}^{n-1} \mathrm{val}(\sigma_i(M)), 0\right).$$

Then, for all integers $a, b \geq 0$ with $a + v < 2b$, any lattice H such that $M_n(\pi^b O_K) \subset H \subset M_n(\pi^a O_K)$ satisfies Eq. (3)

Proof. Recall [4, Definition 3.3] that the *precision polygon* of M is the lower convex hull of the Newton polygons of the entries of $\mathrm{Adj}(X - M)$. By [4, Proposition 3.4], the endpoints of the precision polygon occur at height 0 and $\sum_{i=1}^{n-1} \mathrm{val}(\sigma_i(M))$. By convexity, $O_{K<n}[X] \subset d\chi_M(M_n(\pi^v O_K))$.

Since the coefficients of χ_M are given by polynomials in the entries of M with integral coefficients, [4, Proposition 2.2] implies the conclusion. □

For further discussion of using lattices to measure precision, details on the computation of $d\chi_M$ in (2), and examples of additional precision in χ_M, see [2, §§2.4.3, 3.2.2].

2.2 Several facts about the differential of χ

We may rephrase the condition that $d\chi_M$ is surjective using more familiar invariants attached to M.

Proposition 2.2. *For $M \in M_n(K)$, the following conditions are equivalent:*

(i) the differential $d\chi_M$ is surjective

*(ii) the matrix M has a cyclic vector (*i.e. *M is similar to a companion matrix)*

(iii) the eigenspaces of M over the algebraic closure $\bar{K}$ of K all have dimension 1

(iv) the characteristic polynomial of M is equal to the minimal polynomial of M.

Proof. The equivalence of (ii), (iii), and (iv) is standard; see *e.g.* [9, §7.1]. We now show (ii) ⇒ (i) and (i) ⇒ (iii)

For any $P \in \mathrm{GL}_n(K)$, the image of $d\chi$ at M will be the same as the image of $d\chi$ at PMP^{-1}, so we may assume that M is a companion matrix. For a companion matrix, the last column of $\mathrm{Adj}(X{-}M)$ consists of $1, X, X^2, \ldots, X^{n-1}$ so $d\chi_M$ is surjective.

Now suppose that M has an eigenvalue λ over $\bar{K}$ occurring in multiple Jordan blocks. After conjugating into Jordan normal form over $\bar{K}$, the entries of $\mathrm{Adj}(X{-}M)$ will also be block diagonal, and divisible within each block by the product of $(X - \mu)^{d_\mu}$, where μ, d_μ ranges over the eigenvalues and dimensions of the other Jordan blocks. Since λ occurs in two Jordan blocks, $X - \lambda$ will divide every entry of $\mathrm{Adj}(X{-}M)$ and $d\chi_M$ will not be surjective. □

We also have an integral analogue of Proposition 2.2.

Proposition 2.3. *For $M \in M_n(O_K)$, the following conditions are equivalent:*

(i) the image of $M_n(O_K)$ under $d\chi_M$ is $O_{K<n}[X]$.

(ii) the reduction of M modulo π has a cyclic vector.

Proof. The condition (i) is equivalent to the surjectivity of $d\chi_M$ modulo π. The equivalence with (ii) is as in Proposition 2.2, but over the residue field of K. □

The relationship between precision and the images of lattices under $d\chi_M$ allows us to apply Proposition 2.3 to determine when the precision of the characteristic polynomial is the minimum possible.

Corollary 2.4. *Suppose that $M \in \mathrm{GL}_n(O_K)$ is known with flat precision $O(\pi^N)$. Then the characteristic polynomial of M has precision lattice strictly contained in $O(\pi^N)$ if and only if the reduction of M modulo π does not have a cyclic vector.*

Note that this criterion is checkable using $O\tilde{}(n^{\omega})$ operations in the residue field [17].

Stability under multiplication by X. By definition, the codomain of $d\chi_M$ is $K_{<n}[X]$. However, when M is given, $K_{<n}[X]$ is canonically isomorphic to $K[X]/\chi_M(X)$ as a K-vector space. For our purpose, it will often be convenient to view $d\chi_M$ as a K-linear mapping $M_n(K) \to K[X]/\chi_M(X)$.

Proposition 2.5. *Let R be the subring of $K[X]$ consisting of polynomials f for which $f(M) \in M_n(O_K)$, and $\Lambda = d\chi_M\big(M_n(O_K)\big)$ as a submodule of $K[X]/\chi_M(X)$. Then Λ is stable under multiplication by R.*

Proof. Let $A = \mathrm{Adj}(X{-}M)$ and $f \in R$. By (2), Λ is given by the O_K-span of the entries of A. Using the fact that the product of a matrix with its adjugate is the determinant, $(X{-}M) \cdot A = \chi_M$ and thus $f(X) \cdot A \equiv f(M) \cdot A \pmod{\chi_M(X)}$. The span of the entries of the left hand side is precisely $f(X) \cdot \Lambda$, while the span of the entries of the right hand side is contained within Λ since $f(M) \in M_n(O_K)$. □

Corollary 2.6. *If $M \in M_n(O_K)$, then $d\chi_M\big(M_n(O_K)\big)$ is stable under multiplication by X, so is a module over $O_K[X]$.*

Compact form of $d\chi_M$. Let $\mathscr{C}$ be the companion matrix associated to χ_M as in (1). Assume one of the equivalent conditions of Proposition 2.2, giving a matrix $P \in \mathrm{GL}_n(K)$ such that $M = P\mathscr{C}P^{-1}$. Then Proposition 2.2 also holds for M^T, yielding an invertible matrix $Q \in \mathrm{GL}_n(K)$ with $M^T = Q\mathscr{C}Q^{-1}$.

Proposition 2.7. *We keep the previous notations and assumptions. Let Y be the row vector $(1, X, \ldots, X^{n-1})$. Then*

$$\mathrm{Adj}(X{-}M) = \alpha \cdot PY^T \cdot YQ^T \mod \chi_M \tag{4}$$

for some $\alpha \in K[X]$.

Proof. Write $A = \mathrm{Adj}(X{-}M)$. From $(X{-}M) \cdot A \equiv 0 \pmod{\chi_M}$, we deduce $(X{-}\mathscr{C}) \cdot P^{-1}A \equiv 0 \pmod{\chi_M}$. Therefore each column of $P^{-1}A$ lies in the right kernel of $X{-}\mathscr{C}$ modulo χ_M. On the other hand, a direct computation shows that every column vector W lying in the right kernel of $X{-}\mathscr{C}$ modulo χ_M can be written as $W = w \cdot Y^T$ for some $w \in K[X]/\chi_M$. We deduce that $A \equiv P \cdot Y^T B \pmod{\chi_M}$

for some row vector B. Applying the same reasoning with M^T, we find that B can be written $B = \alpha Y Q^T$ for some $\alpha \in K[X]/\chi_M$ and we are done. □

Proposition 2.7 shows that $\mathrm{Adj}(X-M)$ can be encoded by the datum of the quadruple (α, P, Q, χ_M) whose total size stays within $O(n^2)$: the polynomials α and χ_M are determined by $2n$ coefficients while we need $2n^2$ entries to write down the matrices P and Q. We shall see moreover in Section 4 that interesting information can be read off of this short form (α, P, Q, χ_M).

Remark 2.8. With the previous notation, if $U \in GL_n(K)$, the quadruple for UMU^{-1} is $(\alpha, UP, (U^T)^{-1}Q, \chi_M)$, which can be computed in $O(n^\omega)$ operations in K. This is faster than computing $U\,\mathrm{Adj}(X-M)U^{-1}$ using the dense representation, which requires $\tilde{O}(n^{\omega+1})$ operations in K.

3 HESSENBERG FORM

In this section, we combine the computation of a Hessenberg form of a matrix and the computation of the inverse through the Smith normal form (SNF) over a complete discrete valuation field (CDVF) to compute $\mathrm{Adj}(X-M)$ and $d\chi$. If $M \in M_n(O_K)$, then only division by invertible elements of O_K will occur.

Remark 3.1. In what follows, we will count operations in K (regardless to precision) for expressing our complexities. This choice makes sense because we will not usually need much precision on the entries of $\mathrm{Adj}(X-M)$ (roughly only their valuations matter).

3.1 Hessenberg form

We begin with the computation of an approximate Hessenberg form.

Definition 3.2. A *Hessenberg matrix* is a matrix $M \in M_n(K)$ with

$$M_{i,j} = 0 \text{ for } j \leq i-2.$$

Given integers $n_{i,j}$, an *approximate Hessenberg matrix* is a matrix $M \in M_n(K)$ with

$$M_{i,j} = O(\pi^{n_{i,j}}) \text{ for } j \leq i-2.$$

If $M \in M_n(K)$ and $H \in M_n(K)$ is an (approximate) Hessenberg matrix similar to M, we say that H is an *(approximate) Hessenberg form* of M.

It is not hard to prove that every matrix over a field admits a Hessenberg form. We prove here that we may compute an approximate Hessenberg form for any matrix known at finite jagged precision over K. Moreover, we provide an exact change of basis matrix.

```
Algorithm 1: Approximate Hessenberg form computation
Input: a matrix $M$ in $M_n(K)$.
0. $P := I_n$.  $H := M$.
1. for $j = 1, \dots, n-1$ do
2.   swap the row $j+1$ with a row $i_{min}$ ($i_{min} \geq 2$) s.t. $\mathrm{val}(H_{i_{min},j})$ is minimal.
3.   for $i = j+2, \dots, n$ do
4.     Eliminate the significant digits of $H_{i,j}$ by pivoting with row $j+1$ using a matrix $T$.
5.     $H := H \times T^{-1}$.  $P := T \times P$.
6. Return $H, P$.
```

Note that we use the notation of matrix multiplication in line 5, but actual updating of H and P uses row and column operations (individually using $O(n)$ operations in K).

Proposition 3.3. *Algorithm 1 computes H and P realizing an approximate Hessenberg form of M. P is exact and the computation is in $O(n^3)$ operations in K at precision the maximum precision of a coefficient in M.*

Proof. Inside the nested **for** loop, to eliminate $\pi^{u_y}\varepsilon_y + O(\pi^{n_y})$ with pivot $\pi^{u_x}\varepsilon_x + O(\pi^{n_x})$, the corresponding coefficient of the corresponding shear matrix is any exact element of K congruent to $\pi^{u_y-u_x}\varepsilon_y\varepsilon_x^{-1} \mod \pi^{u_y-u_x \min(n_x-u_x, n_y-u_y)}$. Exactness follows directly. The rest is clear. □

Remark 3.4. From a Hessenberg form of M, it is well known that one can compute the characteristic polynomial of M in $O(n^3)$ operations in K [5, pp. 55–56] However, this computation involves division, and its precision behavior is not easy to quantify.

3.2 Computation of the inverse

In this section, we prove that to compute the inverse of a matrix over K, the Smith normal form is precision-wise optimal in the flat-precision case. We first recall the differential of matrix inversion.

Lemma 3.5. *Let $u : GL_n(K) \to GL_n(K)$, $M \mapsto M^{-1}$. Then for $M \in GL_n(K)$, $du_M(dM) = M^{-1}dMM^{-1}$. It is always surjective.*

We then have the following result about the loss in precision when computing the inverse.

Proposition 3.6. *Let* $\mathrm{cond}(M) = \mathrm{val}(\sigma_n(M))$. *If dM is a flat precision of $O(\pi^N)$ on M then M^{-1} can be computed at precision $O(\pi^{N-2\,\mathrm{cond}(M)})$ by a **SNF** computation and this lower-bound is optimal, at least when N is large.*

Proof. The smallest valuation of an entry of M^{-1} is $-\mathrm{cond}(M)$. Consequently, $N - 2\,\mathrm{cond}(M)$ can be obtained as the valuation of an entry of $du_M(dM)$, and it is the smallest that can be achieved this way for dM in a flat precision lattice. Hence the optimality of the bound given for large N [3, Lemma 3.4].

Now, the computation of the Smith normal form was described in [18]. From M known at flat precision $O(\pi^N)$, we can obtain an exact Δ, and P and Q known at precision at least $O(\pi^{N-\mathrm{cond}(M)})$, with coefficients in O_K and determinant in $O_K^\times$ realizing an Smith normal form of M. There is no loss in precision when computing P^{-1} and Q^{-1}. Since the smallest valuation occurring in Δ^{-1} is $-\mathrm{cond}(M)$, we see that $M^{-1} = Q^{-1}\Delta^{-1}P^{-1}$ is known at precision at least $O(\pi^{N-2\,\mathrm{cond}(M)})$, which concludes the proof. □

3.3 The adjugate of $X-H$

In this section, we compute $\mathrm{Adj}(X-H)$ for a Hessenberg matrix H using the Smith normal form computation of the previous section. The entries of $\mathrm{Adj}(X-H)$ lie in $K[X]$, which is not a complete discrete valuation field, so we may not directly apply the methods of the previous section. However, we may relate $\mathrm{Adj}(X-H)$ to $\mathrm{Adj}(1-XH)$, whose entries lie in $K(\!(X)\!)$. In this way, we compute $\mathrm{Adj}(X-H)$ using an SNF method, with no division in K.

We begin by relating adjugates of similar matrices:

Lemma 3.7. *If $M_1, M_2 \in M_n(K)$ and $P \in GL_n(K)$ are such that $M_1 = PM_2P^{-1}$, then:*

$$\mathrm{Adj}(X - M_1) = P\,\mathrm{Adj}(X - M_2)P^{-1}.$$

The second ingredient we need is reciprocal polynomials. We extend its definition to matrices of polynomials.

Definition 3.8. Let $d \in \mathbb{N}$ and $f \in K[X]$ of degree at most d. We define the reciprocal polynomial of order d of f as $f^{\mathrm{rec},d} = X^d f(1/X)$. Let $F \in M_n(K[X])$ a matrix of polynomials of degree at most d. We denote by $F^{\mathrm{rec},d}$ the matrix with $(F^{\mathrm{rec},d})_{i,j} = (F_{i,j})^{\mathrm{rec},d}$.

We then have the following result :

Lemma 3.9. *Let $M \in M_n(K)$. Then:*

$$\mathrm{Adj}(1 - XM)^{\mathrm{rec},n-1} = \mathrm{Adj}(X - M),$$
$$(\chi_M I_n)^{\mathrm{rec},n} = (1 - XM)\,\mathrm{Adj}(1 - XM).$$

Proof. For any matrix $F \in M_d(K[X])$ of polynomials of degree at most 1, we have $\det(F^{\mathrm{rec},1}) = \det(F)^{\mathrm{rec},d}$. This result directly implies the second part of the lemma; the first part follows from the fact that the entries of $\mathrm{Adj}(X - M)$ and of $\mathrm{Adj}(1 - XM)$ are determinants of size $n - 1$. □

This lemma allows us to compute $\mathrm{Adj}(1-XM)$ instead of $\mathrm{Adj}(X-M)$. This has a remarkable advantage: the pivots during the computation of the SNF of $\mathrm{Adj}(1 - XM)$ are units in $K[[X]]$ (in $O_K[[X]]$ if $M \in M_n(O_K)$) and are known in advance to be on the diagonal. This leads to a very smooth precision and complexity behavior when $M \in M_n(O_K)$.

Algorithm 2: `Approximate` $\mathrm{Adj}(X - H)$
Input: an approximate Hessenberg matrix H in $M_n(O_K)$.

```
0. $U := 1 - XH$. $U_0 := 1 - XH$.
1. While updating $U$, **track** $P$ and $Q$ so that $U_0 = PUQ$ is always satisfied.
2. **for** $i = 1, \dots, n-1$ **do**
3.     **Eliminate**, modulo $X^{n+1}$ the coefficients $U_{i,j}$, for $j \geq i+1$ using the invertible pivot $U_{i,i} = 1 + XL_{i,i} \mod X^{n+1}$ (with $L_{i,i} \in O_K[X]$).
4. **for** $i = 1, \dots, n-1$ **do**
5.     **Eliminate**, modulo $X^{n+1}$ the coefficients $U_{i+1,i}$, using the invertible pivot $U_{i,i}$.
6. $\psi := \prod_i U_{i,i}$.
7. Rescale to get $U = I_n \mod X^{n+1}$.
8. $Y := \psi \times P \times Q \mod X^{n+1}$. [2]
9. **Return** $Y^{\mathrm{rec},n-1}, \psi^{\mathrm{rec},n}$.
```

Theorem 3.10. *Let $H \in M_n(O_K)$ be an approximate Hessenberg matrix with flat precision $O(\pi^N)$. Then, using Algorithm 2, one can compute $\mathrm{Adj}(X-H)$ and χ_H in $\tilde{O}(n^3)$ operations in O_K at precision $O(\pi^N)$.*

Proof. First, the operations of the lines 2 and 3 use $\tilde{O}(n^3)$ operations in O_K at precision $O(\pi^N)$. Indeed, since H is an approximate Heisenberg matrix, when we use $U_{i,i}$ as pivot the only other nonzero coefficient in its column is $U_{i+1,i}$. As a consequence, when performing this column-pivoting, only two rows (i and $i+1$) lead to operations in $O_K[[X]]$ other than checking precision. Hence, line 3 costs $\tilde{O}(n^2)$ for the computation of U. Following line 1, the computation of Q is done by operations on rows, starting from the identity matrix. The order in which the entries of U are cleared implies that Q is just filled in as an upper triangular matrix: no additional operations in $O_K[[X]]$ are required. Thus the total cost for lines 2 and 3 is indeed $\tilde{O}(n^3)$ operations.

For lines 4 and 5, there are only $n-1$ eliminations, resulting in a $\tilde{O}(n^2)$ cost for the computation of U. Rather than actually constructing P, we just track the eliminations performed in order to do the corresponding row operations on Q, since we only need the product $P \times Q$.

Line 6 is in $\tilde{O}(n^2)$ and line 7 in $\tilde{O}(n^3)$.

Thanks to the fact that the P only corresponds to the product of $n-1$ shear matrices, the product on line 8 is in $\tilde{O}(n^3)$. We emphasize that no division has been done. Line 9 is costless, and the result is proven. □

Remark 3.11. If $M \in M_n(K)$ does not have coefficients in O_K, we may apply Algorithms 1 and 2 to $p^v M \in M_n(O_K)$ in $\tilde{O}(n^3)$ operations in O_K, and then divide the coefficient of X^k in the resulting polynomial by p^{kv}.

We will see in Section 5 that for an entry matrix with coefficients known at flat precision, Algorithms 1 and 2 are enough to know the optimal jagged precision on χ_M.

3.4 The adjugate of $X-M$

In this section, we combine Proposition 2.7 with Algorithm 2 to compute the adjugate of $X-M$ when χ_M is squarefree. Note that, under the assumption that $d\chi_M$ is surjective, χ_M is squarefree if and only if M is diagonalizable. The result is the following $\tilde{O}(n^3)$ algorithm, where the only divisions are for gcd and modular inverse computations.

Algorithm 3: `Approximate` $\mathrm{Adj}(X-M)$
Input: an approx. $M \in M_n(O_K)$, with $\mathrm{Disc}(\chi_M) \neq 0$.

```
0. Find $P \in GL_n(O_K)$ and $H \in M_n(O_K)$, approximate Hessenberg, such that $M = PHP^{-1}$, using Algorithm 1.
1. Compute $A = \mathrm{Adj}(X - H)$ and $\chi_M = \chi_H$ using Algorithm 2.
2. Do $\mathrm{row}(A,1) \leftarrow \mathrm{row}(A,1) + \sum_{i=2}^{n} \mu_i \,\mathrm{row}(A,i)$, for random $\mu_i \in O_K$, by doing $T \times A$ for some $T \in GL_n(O_K)$. Compute $B := TAT^{-1}$.
3. Similarly compute $C := S^{-1}BS$ for $S \in GL_n(O_K)$ corresponding to adding a random linear combination of the columns of index $j \geq 2$ to the first column of $B$.
4. **If** $\gcd(C_{1,1}, \chi_M) \neq 1$, **then** go to 2.
5. Let $F$ be the inverse of $C_{1,1} \mod \chi_M$.
6. Let $U := \mathrm{col}(C,1)$ and $Y := F \cdot \mathrm{row}(C,1) \mod \chi_M$.
7. Return $\mathrm{Adj}(X - M) := (PT^{-1}SU \times YS^{-1}TP^{-1}) \mod \chi_M$.
```

Theorem 3.12. *For $M \in M_n(O_K)$ such that $\mathrm{Disc}(\chi_M) \neq 0$, Algorithm 3 computes $\mathrm{Adj}(X - M) \pmod{\chi_M}$ in average complexity*

[2] The product $P \times Q$ should be implemented by sequential row operations corresponding to the eliminations in Step 5 in order to avoid a product of two matrices in $M_n(O_K[X])$.

$\tilde{O}(n^3)$ operations in K. The only divisions occur in taking gcds and inverses modulo χ_M.

Proof. As we have already seen, completing Steps 0 and 1 is in $\tilde{O}(n^3)$. Multiplying by T or S or their inverse corresponds to n operations on rows or columns over a matrix with coefficients in $O_K[X]$ of degree at most n. Thus, it is in $\tilde{O}(n^3)$. Step 5 is in $\tilde{O}(n)$, Step 6 in $\tilde{O}(n^2)$ and Step 7 in $\tilde{O}(n^3)$. We need only prove that the set of P and S to avoid is of dimension at most $n-1$.

The idea is to work modulo $X-\lambda$ for λ a root of χ_M (in an algebraic closure) and then apply Chinese Remainder Theorem. The goal of the Step 2 is to ensure the first row of B contains an invertible entry modulo χ_M. Since $A(\lambda)$ is of rank one, the μ_i's have to avoid an affine hyperplane so that $\mathrm{row}(B,1) \mod (X-\lambda)$ is a non-zero vector. Hence we need only avoid a finite union of affine hyperplanes in order for $\mathrm{row}(B,1) \mod \chi(M)$ to contain an invertible coefficient.

Similarly, the goal of Step 3 is make $C_{1,1}$ invertible modulo χ_M. Again, only a finite union of affine hyperplane need be avoided. Thus, almost any choice of μ_i leads to a matrix C passing Step 4, concluding the proof. □

Remark 3.13. As in the previous section, it is possible to scale $M \in M_n(K)$ so as to get coefficients in O_K and apply the previous algorithm.

Remark 3.14. We refer to [1] for the handling of the precision of gcd and modular inverse computations. In this article, ways to tame the loss of precision coming from divisions are explored, following the methods of [3].

4 FROBENIUS FORM

The algorithm designed in the previous section computes the differential $d\chi_M$ of χ at a given matrix $M \in M_n(K)$ for a cost of $\tilde{O}(n^3)$ operations in K. This seems to be optimal given that the (naive) size of the $d\chi_M$ is n^3: it is a matrix of size $n \times n^2$. It turns out however that improvements are still possible! Indeed, thanks to Proposition 2.7, the matrix of $d\chi_M$ admits a compact form which can be encoded using only $O(n^2)$ coefficients. The aim of this short section is to design a fast algorithm (with complexity $\tilde{O}(n^\omega)$ operations in K) for computing this short form. The price to pay is that divisions in K appear, which can be an issue regarding to precision in particular cases. In this section, we only estimate the number of operations in K and not their behavior on precision.

We now fix a matrix $M \in M_n(K)$ for which $d\chi_M$ is surjective. Let (α, P, Q, χ_M) be the quadruple encoding the short form of $d\chi_M$; we recall that they are related by the relations:

$$d\chi_M(dM) = \mathrm{Tr}(\mathrm{Adj}(X-M)\cdot dM)$$
$$\mathrm{Adj}(X-M) = \alpha \cdot PY^T \cdot YQ^T \mod \chi_M.$$

We may approximate χ_M in $\tilde{O}(n^\omega)$ operations in K by approximating the Frobenius normal form of M [17].

The matrix P can be computed as follows. Pick $c \in K^n$. Define $c_i = M^i c$ for all $i \geq 1$. The c_i's can be computed in $\tilde{O}(n^\omega)$ operations in K, *e.g.* using the first algorithm of [13]. Let P_{inv} be the $n \times n$ matrix whose rows are the c_i's for $1 \leq i \leq n$. Remark that P_{inv} is invertible if and only if $(c_0, c_1, \dots, c_{n-1})$ is a basis of K^n if and only if c is a cyclic vector. Moreover after base change to the basis $(c_0, \dots, c_{n-1})$, the matrix M takes the shape (1). In other words, if P_{inv} is invertible, then $P = P_{\mathrm{inv}}^{-1}$ is a solution of $M = P\mathscr{C}P^{-1}$, where $\mathscr{C}$ is the companion matrix similar to M. Moreover, observe that the condition "P_{inv} is invertible" is open for the Zariski topology. It then happens with high probability as soon as it is not empty, that is as soon as M admits a cyclic vector, which holds by assumption.

The characteristic polynomial χ_M can be recovered thanks to the relation $a_0c_0 + a_1c_1 + \cdots + a_{n-1}c_{n-1} = -c_{n-1} \cdot P$.

Now, instead of directly computing Q, we first compute a matrix R with the property that $\mathscr{C}^T = R\mathscr{C}R^{-1}$. To do so, we apply the same strategy as above except that we start with the vector $e = (1, 0, \dots, 0)$ (and not with a random vector). A simple computation shows that, for $1 \leq i \leq n-1$, the vector $\mathscr{C}^i e$ has the shape:

$$\mathscr{C}^i e = (0, \dots, 0, -a_0, \star, \dots, \star)$$

with $n-i$ starting zeros. Therefore the $\mathscr{C}^i e$'s form a basis of K^n, *i.e.* e is always a cyclic vector of $\mathscr{C}$. We may then let R have columns $e, \mathscr{C}e, \dots, \mathscr{C}^{n-1}e$ and recover Q using the relation $Q = P_{\mathrm{inv}}^T R$.

It remains to compute the scaling factor α. We write

$$\mathrm{Adj}(X-\mathscr{C}) = \alpha \cdot Y^T \cdot YR^T \mod \chi_M \tag{5}$$

by multiplying Eq. (4) on the left by P^{-1} and on the right by P. We observe moreover that the first row of R is $(1, 0, \dots, 0)$. Evaluating the top left entry of Eq. (5), we end up with the relation:

$$\alpha = a_1 + a_2X + \cdots + a_{n-1}X^{n-2} + X^{n-1}.$$

No further computation are then needed to derive the value of α. We summarize this section with the following theorem:

Theorem 4.1. *Given $M \in M_n(K)$ with $d\chi_M$ surjective, one can compute $(\alpha, P, Q, \chi_M) \in K[X]$ with $\tilde{O}(n^\omega)$ operations in K such that $\mathrm{Adj}(X-M) = \alpha \cdot PY^T \cdot YQ^T \mod \chi_M$.*

5 OPTIMAL JAGGED PRECISION

In the previous Sections, 3 and 4, we have proposed algorithms to obtain the adjugate of $X-M$. Our motivation for these computations is to then be able to understand what is the optimal precision on χ_M. In this section, we provide some answers to this question, along with numerical evidence. We also show that it is then possible to derive optimal precision of eigenvalues of M.

5.1 On the characteristic polynomial

For $0 \leq k < n$, let $\beta_k : K[X] \to K$ be the mapping taking a polynomial to its coefficients in X^k. By applying [3, Lemma 3.4] to the composite $\beta_k \circ \chi_M$, one can figure out the optimal precision on the k-th coefficient of the characteristic polynomial of M (at least if M is given at enough precision).

Let us consider more precisely the case where M is given at jagged precision: the (i,j) entry of M is given at precision $O(\pi^{N_{i,j}})$ for some integers $N_{i,j}$. Lemma 3.4 of [3] then shows that the optimal precision on the k-th coefficient of χ_M is $O(\pi^{N'_k})$ where N'_k is given by the formula:

$$N'_k = \min_{1 \leq i,j \leq n} N_{j,i} + \mathrm{val}(\beta_k(A_{i,j})), \tag{6}$$

where $A_{i,j}$ is the (i,j) entry of the adjugate $\mathrm{Adj}(X-M)$.

Proposition 5.1. *If $M \in M_n(\mathcal{O}_K)$ is given at (high enough) jagged precision, then we can compute the optimal jagged precision on χ_M in $\tilde{O}(n^3)$ operations in K.*

Proof. We have seen in §3 and §4 that the computation of the matrix $A = \mathrm{Adj}(X{-}M)$ can be carried out within $\tilde{O}(n^3)$ operations in K (either with the Hessenberg method or the Frobenius method). We conclude by applying Eq. (6) which requires no further operation in K (but n^3 evaluations of valuations and n^3 manipulations of integers). □

Remark 5.2. If $M \in M_n(\mathcal{O}_K)$ is given at (high enough) *flat* precision, then we can avoid the final base change step in the Hessenberg and thus any division by a non-invertible. Indeed, observe that, thanks to Lemma 3.7, we can write:

$$\mathrm{Tr}(\mathrm{Adj}(X{-}M) \cdot dM) = \mathrm{Tr}(\mathrm{Adj}(X{-}H) \cdot P^{-1}dMP)$$

where P lies in $\mathrm{GL}_n(\mathcal{O}_K)$. Moreover, the latter condition implies that $P^{-1}dMP$ runs over $M_n(\mathcal{O}_K)$ when P runs over $M_n(\mathcal{O}_K)$. As a consequence, the integer N'_k giving the optimal precision on the k-th coefficient of M is also equal to $N + \min_{1 \le i,j \le n} \mathrm{val}(\beta_k(A^H_{i,j}))$ where $A^H_{i,j}$ is the (i,j) entry of $\mathrm{Adj}(X{-}H)$, where H is the Hessenberg form of M.

Remark 5.3. As a consequence of the previous discussion, once the optimal jagged precision is known, it is possible to lift the entries of M to a sufficiently large precision, rescale them to have entries in $\mathcal{O}_K$ and then use Algorithm 2 to compute the characteristic polynomial. The output might then need to be rescaled and truncated to the optimal precision. This requires $\tilde{O}(n^3)$ operations in $\mathcal{O}_K$ and may require a large intermediate precision. Better methods for approximating χ_M with lower intermediate precision would be a valuable future contribution.

Numerical experiments. We have made numerical experiments in SageMath [6] in order to compare the optimal precision obtained with the methods explained above with the actual precision obtained by the software. For doing so, we picked a sample of 1000 matrices M in $M_9(\mathbb{Q}_2)$ where all the entries are chosen randomly as follows. We fix an integer N, the relative precision, and generate elements of $\mathbb{Q}_p$ of the shape

$$x = p^v \cdot \left(a + O\!\left(p^{N+v_p(a)}\right)\right)$$

where v is an integer generated according to the distribution:

$$\mathbb{P}[v = 0] = \frac{1}{5} \quad ; \quad \mathbb{P}[v = n] = \frac{2}{5 \cdot |n| \cdot (|n| + 1)} \text{ for } |n| \ge 1$$

and a is an integer in the range $[0, p^N)$, selected uniformly at random. This distribution is chosen merely because it is the default in SageMath.

Once this sample has been generated, we computed, for each $k \in \{0, 1, \dots, 8\}$, the three following quantities:

- the optimal precision on the k-th coefficient of the characteristic polynomial of M given by Eq. (6). (Note that the method for computing $\mathrm{Adj}(X{-}M)$ is irrelevant since we only record the precision loss, not the runtime.)

	Average loss of accuracy		
	Optimal	CR	FP
X^0 (det.)	3.17 dev: 1.76	196 dev: 240	189 dev: 226
X^1	2.98 dev: 1.69	161 dev: 204	156 dev: 195
X^2	2.75 dev: 1.57	129 dev: 164	126 dev: 164
X^3	2.74 dev: 1.73	108 dev: 144	105 dev: 143
X^4	2.57 dev: 1.70	63.2 dev: 85.9	60.6 dev: 85.8
X^5	2.29 dev: 1.66	51.6 dev: 75.3	49.7 dev: 74.9
X^6	2.07 dev: 1.70	9.04 dev: 26.9	8.59 dev: 26.4
X^7	1.64 dev: 1.65	5.70 dev: 15.3	5.38 dev: 14.7
X^8 (trace)	0.99 dev: 1.37	0.99 dev: 1.37	0.99 dev: 1.37

Results for a sample of 1000 instances

Figure 1: Average precision loss on the characteristic polynomial of a random 9×9 matrix over $\mathbb{Q}_2$

- in the capped relative mode (each coefficient carries its own precision which is updated after each elementary arithmetic operation), the precision on the k-th coefficient of the characteristic polynomial of M is computed *via* the call:

  ```
  M.charpoly(algorithm="df").
  ```

  Note that the precision behavior described persists when using algorithms other than the division-free method.
- in the model of floating-point arithmetic (see [2, §2.3]), the number of correct digits of the k-th coefficient of the characteristic polynomial of M.

The table of Figure 1 summarizes the results obtained. It should be read as follows. The acronyms CR and FP refer to “capped relative” and “floating-point” respectively. The numbers displayed are the average loss of *relative* precision. More precisely, if N is the relative precision of the entries of M and v is the valuation of the k-th coefficient of χ_M, then:

- the column “Optimal” is the average of the quantities $(N'_k{-}v) - N$ (where N'_k is defined by Eq. (6)): $N'_k{-}v$ is the optimal *relative* precision, so that the difference $(N'_k{-}v) - N$ is the loss of relative precision;
- the column “CR” is the average of the quatities $(\mathrm{CR}_k{-}v) - N$ where CR_k is the computed (absolute) precision on the k-th coefficient of χ_M;
- the column “FP” is the average of the quatities $(\mathrm{FP}_k{-}v) - N$ where FP_k is the first position of an incorrect digit on the k-th coefficient of χ_M.

We observe that the loss of relative accuracy stays under control in the “Optimal” column whereas it has very erratic behavior in the two other columns, demonstrating the utility of the methods developed in this paper.

5.2 On eigenvalues

Let $M \in M_n(K)$ and $\lambda \in K$ be a *simple* [3] eigenvalue of M. We are interesting in quantifying the optimal precision on λ when M is given with some uncertainty.

To do so, we fix an approximation $M_{\text{app}} \in M_n(K)$ of M and suppose that the uncertainty of M is jagged: each entry of M is given at some precision $O(\pi^{N_{i,j}})$. Let λ_{app} be the relevant eigenvalue of M_{app}. We remark that it is possible to follow the eigenvalue λ_{app} on a small neighborhood $\mathcal{U}$ of M. More precisely, there exists a unique continuous function $f : \mathcal{U} \to K$ such that $f(M_{\text{app}}) = \lambda_{\text{app}}$, and $f(M')$ is an eigenvalue of M' for all $M' \in \mathcal{U}$.

Lemma 5.4. *The function f is strictly differentiable on a neighborhood of M_{app}, with differential at M the map*

$$dM \mapsto d\lambda = -\frac{\operatorname{Tr}(\operatorname{Adj}(\lambda - M) \cdot dM)}{\chi'_M(\lambda)},$$

where χ'_M is the usual derivative of χ_M.

Proof. The first assertion follows from the implicit function theorem, the second from differentiating $\chi_M(\lambda) = 0$, which gives $\chi'_M(\lambda) \cdot d\lambda + \operatorname{Tr}(\operatorname{Adj}(X - M) \cdot dM)(\lambda) = 0$. □

Lemma 3.4 of [3] now implies that, if the $N_{i,j}$'s are large enough and sufficiently well balanced, the optimal precision on the eigenvalue λ is $O(\pi^{N'})$ with:

$$N' = \min_{1 \le i,j \le n} \Big(N_{j,i} + \operatorname{val}(A_{i,j}(\lambda)) - \operatorname{val}(\chi'_M(\lambda)) \Big)$$

where $A_{i,j}$ denotes as above the (i, j) entry of $\operatorname{Adj}(X - M)$. Writing $\operatorname{Adj}(X - M) = \alpha \cdot PY^T \cdot YQ^T \bmod \chi_M$ as in Proposition 2.7, we find:

$$\begin{aligned} N' &= \operatorname{val}(\alpha(\lambda)) - \operatorname{val}(\chi'_M(\lambda)) \\ &\quad + \min_{1 \le i,j \le n} \Big(N_{j,i} + \operatorname{val}(P_i Y(\lambda)^T) + \operatorname{val}(Y(\lambda) Q_j^T) \Big) \end{aligned} \tag{7}$$

where P_i denotes the i-th row of P and, similarly, Q_j denotes the j-th row of Q. Note moreover that $Y(\lambda)$ is the row vector $(1, \lambda, \ldots, \lambda^{n-1})$. By the discussion of §4, the exact value of N' can be determined for a cost of $\tilde{O}(n^\omega)$ operations in K and $O(n^2)$ operations on integers.

When M is given at flat precision, *i.e.* the $N_{i,j}$'s are all equal to some N, the formula for N' may be rewritten:

$$\begin{aligned} N' &= N + \operatorname{val}(\alpha(\lambda)) - \operatorname{val}(\chi'_M(\lambda)) \\ &\quad + \min_{1 \le i \le n} \operatorname{val}(P_i Y(\lambda)^T) + \min_{1 \le j \le n} \operatorname{val}(Y(\lambda) Q_j^T) \Big) \end{aligned} \tag{8}$$

and can therefore now be evaluated for a cost of $\tilde{O}(n^\omega)$ operations in K and only $O(n)$ operations with integers.

To conclude, let us briefly discuss the situation where we want to figure out the optimal jagged precision on a tuple $(\lambda_1, \ldots, \lambda_s)$ of simple eigenvalues. Applying (7), we find that the optimal precision on λ_k is

$$\begin{aligned} N'_k &= \operatorname{val}(\alpha(\lambda_k)) - \operatorname{val}(\chi'_M(\lambda_k)) \\ &\quad + \min_{1 \le i,j \le n} \Big(N_{j,i} + \operatorname{val}(P_i Y(\lambda_k)^T) + \operatorname{val}(Y(\lambda_k) Q_j^T) \Big). \end{aligned}$$

Proposition 5.5. *The N'_k's can be all computed in $\tilde{O}(n^\omega)$ operations in K and $O(n^2 s)$ operations with integers.*

If the $N_{i,j}$'s are all equal, the above complexity can be lowered to $\tilde{O}(n^\omega)$ operations in K and $O(ns)$ operations with integers.

Proof. The $\alpha(\lambda_k)$'s and the $\chi'_M(\lambda_k)$'s can be computed for a cost of $\tilde{O}(ns)$ operations in K using fast multipoint evaluation methods (see §10.7 of [19]). On the other hand, we observe that $P_i Y(\lambda_k)^T$ is the (i, k) entry of the matrix:

$$P \cdot \begin{pmatrix} \lambda_1 & \cdots & \lambda_s \\ \lambda_1^2 & \cdots & \lambda_s^2 \\ \vdots & & \vdots \\ \lambda_1^{n-1} & \cdots & \lambda_s^{n-1} \end{pmatrix}.$$

The latter product can be computed in $\tilde{O}(n^2)$ operations in K. Indeed, the right factor is a truncated Vandermonde matrix, so that computing the above product reduces to evaluating a polynomial at the points $\lambda_1, \ldots, \lambda_s$. Therefore all the $P_i Y(\lambda_k)^T$'s (for i and k varying) can be determined with the same complexity. Similarly all the $Y(\lambda) Q_j^T$ are computed for the same cost. The first assertion of Proposition 5.5 follows. The second assertion is now proved similarly to the case of a unique eigenvalue. □

REFERENCES

[1] Xavier Caruso. Resultants and subresultants of p-adic polynomials. *arxiv:1507.06502*, 2015.

[2] Xavier Caruso. Computations with p-adic numbers. *arxiv:1701.06794*, 2017.

[3] Xavier Caruso, David Roe, and Tristan Vaccon. Tracking p-adic precision. *LMS J. Comput. Math.*, 17(suppl. A):274–294, 2014.

[4] Xavier Caruso, David Roe, and Tristan Vaccon. p-Adic Stability In Linear Algebra. *Proceedings of the 2015 ACM on International Symposium on Symbolic and Algebraic Computation, ISSAC 2015, Bath, United Kingdom, July 06 - 09, 2015*, pages 101–108, 2015.

[5] Henri Cohen. *A course in computational algebraic number theory*, volume 138. Springer Science & Business Media, 2013.

[6] The Sage Developers. *Sage Mathematics Software (Version 7.5)*, 2017. http://www.sagemath.org.

[7] David Harvey. Kedlaya's algorithm in larger characteristic. *International Mathematics Research Notices*, 2007:rnm095, 2007.

[8] David Harvey. Counting points on hyperelliptic curves in average polynomial time. *Annals of Mathematics*, 179(2):783–803, 2014.

[9] Kenneth Hoffman and Ray Kunze. *Linear Algebra.* Prentice-Hall, Englewood Cliffs, New Jersey, 2nd ed. edition, 1971.

[10] Erich Kaltofen. On computing determinants of matrices without division. In P.S. Wang, editor, *Proc. 1992 Internat. Symp. Symbolic Algebraic Computation. (ISSAC'92)*, pages 342–349, New York, 1992. ACM Press.

[11] Erich Kaltofen and Gilles Villard. On the complexity of computing determinants. *Computational Complexity*, 13(3-4):91–130, 2004.

[12] Kiran S. Kedlaya. Counting points on hyperelliptic curves using monsky–washnitzer cohomology. *J. Ramanujan Math. Soc.*, 16:323–338, 2001.

[13] Walter Keller-Gehrig. Fast algorithms for the characteristic polynomial. *Theoretical Computer Science*, 36:309–317, 1985.

[14] François Le Gall. Powers of tensors and fast matrix multiplication. In *Proceedings of the 39th International Symposium on Symbolic and Algebraic Computation*, ISSAC '14, pages 296–303, New York, NY, USA, 2014. ACM.

[15] Clement Pernet and Anre Storjohann. Faster algorithms for the characteristic polynomial. In Dongming Wang, editor, *Proc. 2007 Internat. Symp. Symbolic Algebraic Computation. (ISSAC'07)*, pages 307–314, New York, 2007. ACM Press.

[16] T. R. Seifullin. Computation of determinants, adjoint matrices, and characteristic polynomials without division. *Cybernetics and Systems Analysis*, 38(5):650–672, 2002.

[17] Arne Storjohann. Deterministic computation of the Frobenius form. In *Proceedings of the 42nd IEEE symposium on foundations of computer science*, FOCS '01, pages 368–377, Washington, DC, 2001.

[18] Tristan Vaccon. *p-adic precision.* Theses, Université Rennes 1, July 2015.

[19] Joachim Von Zur Gathen and Jürgen Gerhard. *Modern computer algebra.* Cambridge university press, 2013.

[3] the corresponding generalized eigenspace has dimension 1