Mémoire d'habilitation à diriger des recherches

Université Claude Bernard Lyon I

Numéro d'ordre 17-2003

Algorithmique en algèbre linéaire exacte

Gilles Villard

Chargé de Recherche CNRS

Présenté le 21 mars 2003, après avis des rapporteurs

Philippe Flajolet, INRIA Rocquencourt,
Joachim von zur Gathen, Universität Paderborn,
B. David Saunders, University of Delaware,

et devant le jury composé de

Jean Della Dora, IMAG / INP Grenoble,
Joachim von zur Gathen, Universität Paderborn,
Jean-Michel Muller, CNRS / ENS Lyon,
Jean-Louis Nicolas, Université Claude Bernard Lyon 1,
Luis Miguel Pardo, Universitad de Cantabria,
Marie-Françoise Roy, IRMAR / Université de Rennes 1,
B. David Saunders, University of Delaware.

Travaux menés au Laboratoire de Modélisation et Calcul (CNRS, INPG et UJF) de l'IMAG à Grenoble puis au Laboratoire de l'Informatique du Parallélisme (CNRS, ENS Lyon et INRIA) à Lyon.

Version du 24 mars 2003.

Avant-propos

Ce mémoire traite de la résolution de problèmes de base en algèbre linéaire exacte tels que le calcul du déterminant, de la matrice inverse et de formes normales de matrices. L'orientation est algorithmique et tournée vers des études de complexité, à savoir que l'on se pose la question de calculer les quantités le plus rapidement possible. Outre replacer certains de nos travaux de recherche dans leur contexte, le but en est une synthèse des avancées récentes du domaine.

On sait qu'en modèle algébrique, c'est-à-dire pour des matrices sur un corps K abstrait, les problèmes fondamentaux de l'algèbre linéaire — hormis la résolution de systèmes qui est peut-être plus simple — sont équivalents au produit de matrices (théorème 1.1.1). Le calcul formel transpose la question au cas de matrices à coefficients dans $\mathbb{Z}$ ou dans un anneau de polynômes $\mathsf{K}[x]$. Le fil rouge des recherches en cours et *a fortiori* de ce document, est d'en appréhender la répercussion sur les complexités. On ne s'attaque pas au sujet à brûle-pourpoint, nous allons présenter quelques exemples de réponses partielles auxquelles nous avons contribué. Par contraste avec le cas algébrique où les nombres sont des atomes, on considérait que la répercussion en question était d'ajouter un facteur n relié aux dimensions des matrices et justement dû aux tailles des données à manipuler. On se référera à l'identité (4.1) quant aux longueurs binaires des entiers ou à (5.1) quant aux degrés des polynômes. Nous allons voir des techniques et algorithmes qui convainquent que le surcoût en les dimensions, comparé au cas algébrique, est souvent bien moindre que n (n^η avec $0 \leq \eta < 1$ pour les matrices entières au chapitre 4). Il peut même n'être qu'un facteur logarithmique (déterminant, forme normale de Smith et inversion de matrices polynomiales au chapitre 5). Ces techniques et algorithmes ont aussi des applications dans des contextes plus spécifiques. Nous aborderons le calcul sans division et l'algèbre linéaire des matrices creuses ou plus généralement boîtes noires.

Sans trop anticiper sur les chapitres qui vont suivre, prenons informellement l'exemple du calcul du déterminant d'une matrice A carrée $n \times n$. Sur un corps K abstrait ce calcul est réalisable en $O(n^\omega)$ opérations dans K où ω est l'exposant du produit de matrices. En revanche, si A est à coefficients entiers, la longueur binaire du déterminant est en $O(n \log n \|A\|)$. Appliquer le théorème chinois mène donc à calculer le déterminant en $O(n^\omega \times n \log n \|A\|)$ opérations binaires, et à penser que la complexité est régie par la longueur de la sortie. D'où aussi le facteur n dont nous parlions plus haut. On sait maintenant que ce dernier n'a rien d'essentiel, pourquoi ? Une réponse est que le produit $n^\omega \times n$ n'est qu'une majoration brutale tendant à supposer que de l'ordre de n^ω opérations doivent être effectuées sur des nombres de longueur $O(n \log n \|A\|)$. Il n'en n'est rien et réduire la complexité consiste à faire le maximum d'opérations sur des entiers courts pour limiter l'apparition des entiers plus longs aux phases finales des algorithmes (et à la sortie nécessairement). Concrètement cela va s'illustrer avec des algorithmes "par blocs" comportant des grandes phases d'augmentation (ou de diminution) des longueurs mises en jeu.

Le déterminant se révèle être le cas le "plus simple", peut-être parce que de son calcul comme produit des facteurs invariants (forme normale de Smith) se dégagent naturellement des phases de calcul associées à ces facteurs. Nous présentons plusieurs nouveaux algorithmes aux chapitres 3

et 4. Les résultats sont plus difficiles (encore partiels) pour des problèmes tels que le polynôme caractéristique ou l'inversion. Nous verrons néanmoins au chapitre 5 quelques progrès dans le cas des matrices polynomiales. Le surcoût est alors dû aux degrés des polynômes plutôt qu'aux longueurs des entiers et les raisonnements sont très analogues.

Le "lien" entre ces calculs sur $\mathbb{Z}$ ou sur $\mathsf{K}[x]$ et le calcul sans division vient du procédé d'élimination des divisions de Strassen. Il fait en effet intervenir des séries formelles et l'ordre auquel on les tronque détermine une longueur qui joue sur les coûts. Le rapport entre les premiers chapitres et le chapitre 6 sur les matrices boîtes noires est plus clair. L'utilisation de l'approche Krylov / Lanczos est en effet un outil important aussi bien pour des matrices denses que pour ces dernières.

Le mémoire est organisé comme suit. Les problèmes concernés sont introduits au chapitre 1 avec les définitions et rappels de base pour le reste de la lecture. Un paragraphe y propose aussi un bref panorama des méthodes de résolution. Le chapitre 2 se consacre ensuite à l'approche Krylov / Lanczos. Communément utilisée en analyse numérique, elle apporte aussi des progrès notables en calcul exact et en complexité. En particulier, le chapitre 3 montre comment elle permet d'améliorer la complexité du calcul du déterminant sans division. Comme annoncé, les chapitres 4 et 5 détaillent ensuite des aspects nouveaux du calcul sur les matrices à coefficients entiers ou polynomiaux. Le chapitre 6 s'intéresse aux algorithmes spécialisés pour les boîtes noires et à la notion de pré-conditionnement algébrique.

Modèle de calcul, algorithmes probabilistes. Nous restons informel sur le modèle de calcul. Disons que les algorithmes déterministes correspondent à des programmes d'évaluation et à des machines Ram voire Pram, binaires et arithmétiques sur un anneau R ou un corps K abstraits. Pour les algorithmes probabilistes nous supposons aussi disposer, pour une entrée de longueur l du programme, d'un nombre polynomial en l de bits aléatoires ou d'éléments aléatoires d'un sous-ensemble fini de l'anneau ou du corps. Deux types d'algorithmes probabilistes sont manipulés. Un algorithme de type Las Vegas — toujours correct et probablement rapide — retourne une réponse juste pour une proportion constante des entrées aléatoires et retourne "échec" sinon. Un algorithme de type Monte Carlo — toujours rapide et probablement correct — retourne la bonne réponse pour une proportion constante des entrées aléatoires et n'est pas spécifié sinon.

Les coûts des algorithmes sont le plus souvent donnés en termes de "$\tilde{O}(\ )$" où $\tilde{O}(f(n))$ signifie $O(f(n)\log^{O(1)} f(n))$. Le coût du produit de deux polynômes de degrés n dans $\mathsf{R}[x]$ est noté $\mathrm{M}(n)$. On sait que l'on peut prendre $\mathrm{M}(n) = O(n \log n \log\log n)$ [20]. L'exposant du produit de matrices est noté ω (cf §1.1). Le degré d'une matrice sur $\mathsf{K}[x]$, fréquemment noté d, est le maximum des degrés de ses coefficients. Cela amène des complexités en fonction des dimensions des matrices et de d. Pour une matrice A dans $\mathbb{Z}^{n\times n}$, la norme infinie est notée $\|A\|$. Si $a_{i,j}$ est le coefficient i,j alors $\|A\| = \max_{1\le i\le n} \sum_{j=1}^{n} |a_{i,j}|$ et chaque coefficient de A s'écrit avec un nombre de bits borné par $1 + \log(\|A\| + 1)$. Les complexités correspondantes sont données en fonction de n et de $\|A\|$.

Ouvrages de référence. Le livre de Gantmacher [49] est une référence fondamentale pour tous les aspects d'algèbre que nous regardons. Comme autres ouvrages généralistes en algèbre linéaire, notamment concernant les formes normales, citons aussi MacDuffee [86] et Newman [97].

L'ouvrage de von zur Gathen & Gerhard [51] fait le tour des techniques algorithmiques essentielles du calcul formel. Sur le sujet on se reportera aussi au volume encyclopédique [24]. Pour la complexité algébrique on pourra se référer à Bürgisser, Clausen & Shokrollahi [18]. Quant à l'algorithmique exacte des matrices, le livre de Bini & Pan [12] permet un large survol en modèles

algébriques ou binaires. Celui de Abdeljaoued & Lombardi [2] focalise plus particulièrement sur le polynôme caractéristique en algébrique. Sur la question des descriptions de fractions et des matrices polynomiales nous nous référons à Kailath [64] ainsi qu'à Gohberg, Lancaster & Rodman [59].

Table des matières

1. Questions de base et formes normales

1.1 Déterminant, rang, noyau

Le coût du produit de matrices sert en général de repère pour comparer les coûts des algorithmes de base en algèbre linéaire. Il représente même souvent une borne inférieure. On introduit donc un paramètre $\omega \in [2, 3]$ pour désigner l'exposant du produit de deux matrices carrées $n \times n$ sur un corps commutatif K. Ce paramètre représente toute valeur telle que le produit puisse être calculé en $O(n^\omega)$ additions, soustractions ou multiplications dans K, voire la meilleure valeur possible [18, §15.1 et §16.1]. Le produit de matrices habituel donne $\omega = 3$ et l'algorithme de Coppersmith et Winograd conduit à $\omega \approx 2.376$ [28]. Des exposants spécifiques au produit de matrices rectangulaires sont donnés dans [27, 62].

Parmi les premières questions qui se posent après le produit de matrices, pour une matrice A de dimensions $n \times m$ donnée sur un corps, sont la résolution de systèmes linéaires

$$Ay = b,\ b \in \mathsf{K}^n, \tag{1.1}$$

le calcul du rang r de la matrice ainsi que le calcul d'une base N de son noyau

$$AN = 0,\ N \in \mathsf{K}^{m\times(m-r)}. \tag{1.2}$$

Si A est carrée $n \times n$, on cherche à calculer son déterminant $\det A$, son polynôme caractéristique $\det(xI - A) \in \mathsf{K}[x]$ et son inverse. Les nombreuses études sur le sujet montrent que pour un corps K abstrait, toutes ces questions sont extrêmement voisines. On se référera aux récapitulatifs dans [12, Chap. 2] ou [18, Chap. 16].

Théorème 1.1.1. *Pour une matrice inversible sur un corps K abstrait la solution à un système linéaire, le déterminant et l'inverse se calculent en $O(n^\omega)$ opérations dans K [118, 110, 17]. Le polynôme caractéristique se calcule en $O(n^\omega \log n)$ opérations dans K [78].*
En calculs par circuits arithmétiques, l'exposant ω est le meilleur possible pour le déterminant [4], pour l'inverse [119, 17] et pour le calcul du polynôme caractéristique (puisque son terme constant donne le déterminant).

On ne sait pas si la résolution de système linéaire est un problème de même complexité ou plus simple que le produit de matrices [4, Question p. 328].

Pour une matrice à coefficients sur un anneau principal R comme $\mathbb{Z}$ ou $\mathsf{K}[x]$ on peut se poser les mêmes questions (1.1) et (1.2) sur le corps $\mathbb{Q}$ ou $\mathsf{K}(x)$ associé. On peut aussi chercher à résoudre les problèmes diophantiens correspondants c'est-à-dire en restant dans l'anneau pour

$$Ay = b,\ b \in \mathsf{R}^n, \tag{1.3}$$

et pour

$$AM = 0,\ M \in \mathsf{R}^{m\times(m-r)}. \tag{1.4}$$

Dans ce dernier cas, puisqu'il n'y a pas unicité, on peut se donner des contraintes supplémentaires et calculer la base formée par les colonnes de M sous une forme particulière, comme par exemple sous forme de base minimale (voir §1.4).

En dehors des cas purement algébriques, les coûts de résolution de ces problèmes sont *a priori* reliés aux tailles des données manipulées. Il s'agira de longueurs d'entiers (en nombres de bits) et de degrés de polynômes. La première taille qui peut avoir un impact est celle du déterminant. Dans le cas entier, la borne d'Hadamard [51, Theorem 16.6]

$$\log|\det A| \leq (n/2)\log n + n\log\|A\|,\ A \in \mathbb{Z}^{n\times n} \tag{1.5}$$

indique que le déterminant peut comporter jusqu'à $\tilde{O}(n\log\|A\|)$ chiffres. Dans le cas polynomial, pour A de degré d, on a immédiatement

$$\deg\det A \leq nd,\ A \in \mathsf{K}[x]^{n\times n}.$$

Ni sur $\mathsf{K}[x]$ en comptant les opérations dans K, ni en complexité binaire sur $\mathbb{Z}$ en comptant le nombre d'opérations sur les bits, on ne dispose d'analogue au théorème 1.1.1 de complexité algébrique. Les difficultés relatives des problèmes sont méconnues.

1.2 Forme normale de Smith

Plus précisément que par le déterminant, le lien entre les tailles des données et les coûts des algorithmes pourra être révélé par la forme normale de Smith [111, 97] (voir la fin de l'introduction au chapitre 4 en page 27).

Théorème 1.2.1. *Toute matrice $n \times m$ sur un anneau principal R est équivalente à une matrice diagonale*

$$S = UAV = \begin{bmatrix} s_1 & & & & \\ & s_2 & & & \\ & & \ddots & & \\ & & & s_r & \\ & & & & \end{bmatrix} \in \mathsf{R}^{n\times m}$$

où U et V sont unimodulaires sur R (carrées et de déterminants inversibles dans R) ; où les seuls coefficients non nuls sont les r coefficients diagonaux s_i, appelés facteurs invariants de A ; où ces facteurs invariants vérifient $s_{i+1}|s_i$, $1 \leq i \leq r-1$. Cette matrice diagonale est appelée forme normale de Smith de A. Elle est unique à multiplication près des s_i par des éléments inversibles de l'anneau.

Une première intervention de cette forme concerne la résolution de systèmes linéaires. Pour $A \in \mathsf{R}^{n\times n}$ de déterminant non nul et $b \in \mathsf{R}^n$, supposons que les fractions composantes de y dans (1.1) soient réduites. Alors, au delà des règles de Cramer, on sait que le plus petit dénominateur commun de ces composantes est au plus s_1, le plus grand facteur invariant de A (voir aussi la notion de matrice adjointe "réduite" [49, Chap. 4, §6]).

Exemple. On a par exemple

$$S = UAV = \begin{bmatrix} 3 & 2 & 1 \\ 6 & 3 & 1 \\ 19 & 9 & 3 \end{bmatrix} \begin{bmatrix} 19 & -44 & 0 \\ -57 & 134 & -6 \\ 51 & -124 & 18 \end{bmatrix} \begin{bmatrix} 0 & 2 & 7 \\ 0 & 1 & 3 \\ 1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 6 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

La solution d'un système $Ay = b$ est alors $y = VS^{-1}Ub$. Donc seule la matrice inverse de S donne au plus 6 comme dénominateur à y. □

Nous aurons l'occasion de voir en détail l'influence de la forme dès qu'il s'agira de calculer le déterminant ou le polynôme caractéristique. En effet, par construction, pour A inversible on a

$$\det A = s_1 s_2 \dots s_n. \tag{1.6}$$

La forme normale est génériquement triviale, c'est-à-dire qu'en général, si A n'est pas inversible ses facteurs invariants sont tous égaux à 1 ; si A est inversible, tous sauf le premier (qui est égal au déterminant) sont égaux à 1. Nous avons par exemple montré le résultat suivant avec W. Eberly et M. Giesbrecht.

Théorème 1.2.2. *Soit $a \in \mathbb{Z}$ et soit $\Lambda = \{a, a+1, \dots, a+\lambda-1\}$ un ensemble de λ entiers consécutifs. Si A est une matrice entière $n \times n$ dont les coefficients sont choisis au hasard uniformément et indépendamment dans Λ alors le nombre moyen de facteurs invariants non triviaux de A est en $O(\log_\lambda n)$ [40, Corollary 6.3].*

1.3 Forme réduite et forme normale de Popov

Nous utiliserons la forme normale de Popov pour la normalisation de matrices sur $\mathsf{K}[x]$ vues soit comme matrices polynomiales (pour les bases minimales, §1.4) soit comme polynômes matriciels (pour les polynômes minimaux, §1.5). Cette notion, courante en calcul algébrique, est l'analogue plus simple de la réduction de bases de réseaux entiers (voir par exemple [50] et [51, Exercise 16.12]). Nous avons repris le nom de Popov qui a utilisé cette forme pour normaliser les systèmes linéaires en théorie du contrôle [103, 64, 124].

Ici, pour une matrice A dans $\mathsf{K}[x]^{n\times m}$ ce sont les degrés maximaux d_j, $1 \le j \le m$, des colonnes prises séparément qui importent. Soit $A_t \in \mathsf{K}^{n\times m}$ la matrice dont la colonne j est le vecteur des coefficients de x^{d_j} dans la colonne j de A.

Théorème 1.3.1. *Une matrice polynomiale A de rang r est dite réduite en colonnes si sa "matrice de tête" A_t possède r colonnes indépendantes et $m-r$ colonnes nulles. Pour toute matrice $B \in \mathsf{K}[x]^{n\times m}$ il existe une matrice unimodulaire $U \in \mathsf{K}[x]^{m\times m}$ telle que BU soit réduite en colonnes [64, §6.3.2].*

S'il y a besoin d'unicité on dispose d'une forme réduite en colonnes normalisée dite sous forme normale de Popov. On choisit de prendre la matrice de tête A_t sous forme échelonnée en colonnes avec des éléments pivots égaux à 1 (les polynômes correspondants dans A sont unitaires). Les polynômes dans une ligne de A sont de degrés strictement plus petits que celui de l'élément correspondant au pivot dans la même ligne de A_t. On reconnaît là la normalisation habituellement utilisée pour la forme normale d'Hermite [97]. Nous avons proposé avec B. Beckermann et G. Labahn une définition de forme normale à décalages qui entre autres unifie les deux cas [8, 7]. D'après [64, §6.7.2], pour toute matrice polynomiale B il existe une matrice unimodulaire U telle que BU soit sous forme de Popov.

La forme normale de Popov en lignes se construit de manière analogue, par exemple à partir de la forme en colonnes de la matrice transposée.

1.4 Bases minimales et indices de Kronecker

Quitte à ré-arranger ses colonnes, si A sur $\mathsf{K}[x]$ est réduite en colonnes, on supposera que les degrés $d_1, d_2, \ldots, d_r$ sont ceux de ses colonnes non nulles et qu'ils vérifient $d_1 \leq d_2 \leq \ldots \leq d_r$. Une propriété importante est la minimalité de ces degrés.

Théorème 1.4.1. *Soit* M *le* $\mathsf{K}[x]$*-module engendré par les colonnes de* A*. Si* A *est réduite en colonnes de degrés* $d_1, \ldots, d_r$ *alors les degrés* d'_j *rangés en ordre croissant d'une base quelconque de* M *satisfont à* $d'_j \geq d_j$*,* $1 \leq j \leq r$ *[64, §6.5.4]. On dit que les colonnes de* A *forment une base minimale de* M*.*

Par conséquent, deux matrices réduites en colonnes et (unimodulairement) équivalentes à droite ont les mêmes degrés (en colonnes à l'ordre près). Quand M est l'espace nul d'une matrice, alors ces degrés sont appelés indices de Kronecker (à droite) de la matrice dont ils caractérisent la "singularité" [49].

Exemple. On a

$$AM = \begin{bmatrix} x & 1 & 0 & 0 & 0 & 0 \\ 0 & x & 1 & 0 & 0 & 0 \\ 0 & 0 & x & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} -1 & 0 & 0 \\ x & 0 & 0 \\ -x^2 & 0 & 0 \\ x^3 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = 0.$$

Puisque M est réduite en colonnes de degrés $3, 0, 0$ et forme une base du noyau de A (comme module), ces degrés sont les indices de Kronecker de A (on a ici un faisceau $A_0 + xA_1$). Toute autre base dans $\mathsf{K}[x]^{6\times 3}$ du noyau de A comportera au moins un vecteur de degré 3. □

Les valeurs de ces degrés minimaux interviennent directement dans les coûts de certains algorihmes. On ne dispose en général que d'une borne sur leur somme avec notamment un transfert des indices structuraux. On définit pour cela $\delta(A)$ le degré de McMillan d'une matrice polynomiale A de rang r comme le maximum des degrés des déterminants de ses sous-matrices $r \times r$ [65]. Pour $A \in \mathsf{K}[x]^{n\times m}$ de degré d, dont le module des colonnes a comme degrés minimaux $d_1, \ldots, d_r$ et d'indices de Kronecker à droite $\beta_1, \ldots, \beta_{m-r}$ on a

$$\delta(A) = \sum_{j=1}^{r} d_j + \sum_{l=1}^{m-r} \beta_l \tag{1.7}$$

et en particulier,

$$\sum_{l=1}^{m-r} \beta_l \leq rd - \sum_{j=1}^{r} d_j.$$

Nous avons donné une version plus générale de ces identités pour borner les degrés des matrices de passage vers les formes normales à décalages [8, 7].

1.5 Polynômes minimaux

La notion de polynôme minimal sera essentielle à l'approche Krylov / Lanczos. Le polynôme minimal d'une suite de matrices $(A_i)_{i\geq 0}$ sur K engendrée par une récurrence linéaire scalaire

$$A_i f_0 + A_{i+1} f_1 + \ldots + A_{i+d} f_d = 0,\ i \geq 0,\ f_j \in \mathsf{K}, \tag{1.8}$$

est le polynôme unitaire $f_0 + f_1 x + \ldots + f_{d-1} x^{d-1} + x^d$ de plus petit degré donnant une relation de récurrence (1.8) possible. Dans le cas d'une récurrence matricielle à droite définie par

$$A_i F_0 + A_{i+1} F_1 + \ldots + A_{i+d} F_d = 0,\ i \geq 0,\ F_j \in \mathsf{K}^{m\times m}, \tag{1.9}$$

et à l'image du cas scalaire on peut considérer les polynômes matriciels du type

$$F(x) = F_0 + F_1 x + \ldots + F_d x^d \in \mathsf{K}[x]^{m\times m}. \tag{1.10}$$

Pour une suite donnée on vérifie que les colonnes des polynômes matriciels vérifiant (1.9) forment un $\mathsf{K}[x]$-sous-module M_{A_i} de $\mathsf{K}[x]^m$. Nous avons donc proposé la définition suivante.

Définition 1.5.1. *On appelle polynôme minimal matriciel (à droite) d'une suite de matrices donnée par une récurrence linéaire du type (1.9), un polynôme matriciel correspondant comme en (1.10), sous forme réduite en colonnes et dont les colonnes forment une base de* M_{A_i}. *On parle* du *polynôme minimal pour l'unique polynôme minimal sous forme normale de Popov [127].*

Vu le choix de normalisation pour la forme de Popov (pivots égaux à 1), le polynôme minimal matriciel coïncide avec le polynôme minimal dans le cas (1.8).

Exemple. Soit la suite de matrices

$$(A_i)_{i\geq 0} = \left(\begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix}, \ldots \right)$$

sur $\mathbb{Z}/2\mathbb{Z}$, de laquelle le polynôme matriciel

$$M(x) = \begin{bmatrix} x^2+x+1 & 1+x^2 \\ 0 & 1 \end{bmatrix}$$

est polynôme générateur. On a par exemple,

$$A_0 M_0 + A_1 M_1 + A_2 M_2 = \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} + \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}\begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix}\begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix} = 0.$$

On peut vérifier que les colonnes de M engendrent le module M_{A_i} correspondant. La matrice $M(x)$ est équivalente en colonnes à la matrice

$$F(x) = \begin{bmatrix} x & 1 \\ 1 & x+1 \end{bmatrix}$$

qui est sous forme de Popov. Cette dernière est donc le polynôme minimal matriciel de la suite, en particulier

$$A_0 F_0 + A_1 F_1 = \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} + \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = 0.$$

□

Proposition 1.5.2. *Une suite vérifie une récurrence scalaire (1.8) si et seulement si elle vérifie une récurrence matricielle (1.9) [121, Theorem 4.1].*

Démonstration. Cette dernière condition est suffisante puisque si $F \in \mathsf{K}[x]^{m\times m}$ engendre la suite selon (1.9) alors c'est aussi le cas pour FF^* où F^* est la matrice adjointe de F. Or, $FF^* = \operatorname{diag}(\det F, \det F, \ldots, \det F)$, donc $\det F \in \mathsf{K}[x]$ engendre la suite selon (1.8). Réciproquement, si $f \in \mathsf{K}[x]$ engendre la suite selon (1.8) alors $\operatorname{diag}(f, f, \ldots, f)$ engendre la suite matriciellement. □

Le problème de calculer le polynôme minimal d'une suite donnée a été très étudié aussi bien dans le cas scalaire que matriciel. Outre l'approche Berlekamp / Massey, différents point de vue peuvent être choisis comme la résolution de systèmes linéaires structurés, l'approximation de Padé ou l'algorithme d'Euclide. Rappelons que l'on note $\mathrm{M}(d)$ le nombre d'opérations arithmétiques du produit de deux polynômes de degrés d dans $\mathsf{K}[x]$.

Théorème 1.5.3. *S'il est de degré au plus d donné, le polynôme minimal $F \in \mathsf{K}[x]^{m\times m}$ d'une suite de matrices (1.9) linéairement engendrée dans $\mathsf{K}^{m\times m}$ se calcule en $O(m^3d^2)$ ou, avec transformée de Fourier rapide, en $O(m^2\mathrm{M}(md)\log(md))$ opérations dans K à partir des $2d$ premiers termes de la suite.*

Pour une introduction à l'algorithmique correspondante on peut se référer à [51, §12.3]. Quand $m = 1$, le coût $O(\mathrm{M}(d)\log(d))$ correspond à l'algorithme de Knuth / Schönhage [79, 109] pour l'algorithme d'Euclide sur des entiers. La même approche donne une version rapide de l'algorithme de Berlekamp / Massey [89]. Quant à l'aspect approximation de Padé et résolution de systèmes Toeplitz ou Hankel voir aussi [13, 15, 92]. Dans le cas matriciel, pour le calcul d'un polynôme minimal non forcément normalisé, l'algorithme de Berckermann & Labahn part du point de vue de l'approximation [5]. Celui de Coppersmith généralise plus directement la technique de Berlekamp / Massey [26]. On verra aussi [58] ou [67] et ses références autour des matrices structurées. Calculer le polynôme minimal sous forme normale de Popov revient à inclure une normalisation au fil des algorithmes matriciels précédents. C'est par exemple l'algorithme Fffg de Beckermann & Labahn en rapport avec l'approximation de Padé [6, 8] ou ce que nous avons nous-mêmes proposé en utilisant des matrices structurées [124].

Nous spécialiserons le résultat du théorème 1.5.3 pour des suites "normales" au paragraphe 2.5. Dans [80], nous avons eu l'occasion avec P. Koiran et N. Portier de travailler sur les polynômes minimaux de suites "q-récurrentes". Une suite $(A_i)_{i\geq 0}$ de vecteurs de K^n est dite q-récurrente d'ordre d, pour q lui-même vecteur de K^n, si la suite $(q^T A_i)_{i\geq 0}$ est linéairement récurrente.

1.6 Fractions et pgcd de matrices

Une matrice $A \in \mathsf{K}(x)^{n\times m}$ de fractions rationnelles peut se décrire à l'aide d'un couple — appelé description à droite — formé d'une matrice $P \in \mathsf{K}[x]^{n\times m}$ et d'une matrice inversible $Q \in \mathsf{K}[x]^{m\times m}$ telles que $A = PQ^{-1}$. Comme dans le cas scalaire, on va voir que l'ensemble des descriptions possibles d'une fraction donnée s'obtient à partir d'une description irréductible.

Définition 1.6.1. *Une fraction $A \in \mathsf{K}(x)^{n\times m}$ est dite strictement propre si $\lim_{x\to\infty} A(x) = 0$.*

On peut alors facilement vérifier le lemme qui suit.

Lemme 1.6.2. *Si (P, Q) est une description à droite de A et que A est strictement propre, alors les degrés des colonnes de P sont strictement inférieurs aux degrés des colonnes correspondantes de Q.*

Pour arriver à la notion d'irréductibilité, la forme normale de Popov en lignes nous permet d'abord de définir un plus grand commun diviseur à droite unique pour les polynômes matriciels. Avec P et Q comme ci-dessus, soit $U \in \mathsf{K}[x]^{(m+n)\times(m+n)}$ unimodulaire telle que

$$U(x) \begin{bmatrix} P(x) \\ Q(x) \end{bmatrix} = \begin{bmatrix} G(x) \\ 0 \end{bmatrix} \in \mathsf{K}[x]^{(m+n)\times m} \tag{1.11}$$

avec $G \in \mathsf{K}[x]^{m\times m}$ sous forme normale de Popov. Puisque Q est inversible, G l'est aussi. On voit à partir de

$$U(x)^{-1} \begin{bmatrix} G(x) \\ 0 \end{bmatrix} = \begin{bmatrix} P(x) \\ Q(x) \end{bmatrix}$$

que G est un diviseur commun à droite de P et de Q. En effet, en choisissant pour $U_1 \in \mathsf{K}[x]^{n\times m}$ et $U_2 \in \mathsf{K}[x]^{m\times m}$ les sous-matrices adéquates de U^{-1}, on a

$$U_1(x)G(x) = P(x),\ U_2(x)G(x) = Q(x). \tag{1.12}$$

Comme énoncé dans le théorème suivant la matrice G est même maximale parmi tous les diviseurs communs.

Théorème 1.6.3. *Le polynôme matriciel G est un plus grand diviseur commun à droite de P et de Q, c'est-à-dire que tout autre diviseur polynomial M commun à droite est diviseur à droite de G. En d'autres termes, si*

$$P(x) = S(x)M(x) \text{ et } Q(x) = T(x)M(x)$$

pour S et T matrices sur $\mathsf{K}[x]$ alors il existe une matrice polynomiale V telle $G = VM$ [64, §6.3]. Puisque G est définie de manière unique à partir de P et de Q on peut considérer que G est le *pgcd (à droite) de P et de Q.*

Toute matrice VG pour V unimodulaire est un plus grand commun diviseur à P et à Q. On voit à partir du théorème 1.4.1 (en lignes) que le choix de la forme de Popov pour la normalisation conduit à un pgcd avec des degrés de lignes minimaux.

Les deux matrices P et Q avec Q inversible sont dites premières entre elles si leur pgcd est l'identité, on dit alors qu'elles forment une description irréductible de $A = PQ^{-1}$. On aura aussi l'occasion d'utiliser la caractérisation suivante.

Corollaire 1.6.4. *Si P et Q sont premières entre elles alors il n'existe pas de x_0 tel que $P(x_0)$ et $Q(x_0)$ aient un vecteur non nul en commun dans leurs noyaux.*

En effet, s'il existait un tel vecteur $u \neq 0$, d'après (1.11) on aurait

$$U(x_0) \begin{bmatrix} P(x_0) \\ Q(x_0) \end{bmatrix} u = \begin{bmatrix} I \\ 0 \end{bmatrix} u = 0,$$

ce qui n'est pas possible.

Exemple. Le corollaire porte bien sur l'existence de vecteurs propres en commun, l'existence de valeurs propres communes ne dit pas grand chose. Considérons par exemple

$$P(x) = \begin{bmatrix} x-1 & 0 \\ 0 & x+1 \end{bmatrix} \text{ et } Q(x) = \begin{bmatrix} x+1 & 0 \\ 0 & x-1 \end{bmatrix}.$$

Ces deux matrices ont un pgcd égal à l'identité. Un pgcd non trivial signifie au moins un zéro en commun mais aussi que la structure propre associée à ce zéro soit commune. Par exemple

$$P(x) = \begin{bmatrix} x-1 & 0 \\ 0 & x+1 \end{bmatrix} \text{ et } Q'(x) = \begin{bmatrix} x+2 & x+1 \\ 0 & x^2-1 \end{bmatrix}$$

dont le pgcd est

$$G(x) = \begin{bmatrix} 1 & 0 \\ 0 & x+1 \end{bmatrix}.$$

admettent -1 comme zéro commun et

$$P(-1) = \begin{bmatrix} -2 & 0 \\ 0 & 0 \end{bmatrix} \text{ et } Q'(-1) = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$$

ont un noyau commun donné par le vecteur $[0 \;\; 1]^T$. □

Théorème 1.6.5. *Soit une description $A = PQ^{-1} \in \mathsf{K}[x]^{n\times m}$ irréductible de A. Pour toute autre description $A = ST^{-1}$ il existe une matrice polynomiale $G \in \mathsf{K}[x]^{m\times m}$ telle que $S = PG$ et $T = QG$ (G est un pgcd de S et de T). Si S et T sont aussi premières entre elles alors G est unimodulaire [64, Theorem 6.5-4].*

1.7 Similitude et forme normale de Frobenius

Au paragraphe 1.2 nous avons évoqué l'importance de la forme normale de Smith pour le calcul du déterminant. La forme normale de Frobenius [47] que nous allons maintenant définir, jouera un rôle analogue pour le calcul du polynôme caractéristique χ_A. Ce dernier est en effet un cas particulier de déterminant puisque $\chi_A(x) = \det(xI - A)$. La forme de Frobenius fait de même intervenir le polynôme minimal π_A de la matrice, elle aura aussi son importance pour la compréhension de l'approche Krylov / Lanczos par blocs.

Théorème 1.7.1. *Toute matrice $A \in \mathsf{K}^{n\times n}$ est semblable à une unique matrice diagonale par blocs*

$$F = P^{-1}AP = \operatorname{diag}\left(C_{f_1}, C_{f_2}, \ldots, C_{f_\phi}\right) \in \mathsf{K}^{n\times n}$$

où les blocs C_{f_i}, $1 \le i \le \phi$, sont des matrices compagnons définies par des polynômes $f_i \in \mathsf{K}[x]$ vérifiant $f_{i+1} | f_i$, $1 \le i \le \phi - 1$, et appelés facteurs invariants de A. Cette matrice est appelée forme normale de Frobenius de A [49, Chap. 6, §5].

En particulier, le "plus grand facteur invariant" f_1 (de plus haut degré) est le polynôme minimal de A. Le terme "facteur invariant" a le même sens qu'au paragraphe 1.2.

Théorème 1.7.2. *La forme normale de Frobenius de A est* $\operatorname{diag}\left(C_{f_1}, C_{f_2}, \ldots, C_{f_\phi}\right)$ *si et seulement si la forme normale de Smith de $xI - A$ est* $\operatorname{diag}(f_1, f_2, \ldots, f_\phi, 1, \ldots, 1)$ *[49, Chap. 6, §5-§6].*

En particulier, comme pour (1.6) on a :

$$\chi_A(x) = f_1(x) f_2(x) \ldots f_\phi(x). \tag{1.13}$$

Les liens théoriques entre les deux formes normales ont conduit à beaucoup d'analogies entres les algorithmes pour les calculer, de même que certains résultats reposent sur une approche hybride équivalence unimodulaire / similitude (voir [126, 117] et [114, Chap. 9]).

Giesbrecht a donné un algorithme probabiliste Las Vegas pour calculer la forme de Frobenius en $\tilde{O}(n^\omega)$ opérations dans K [53]. Cette complexité est atteinte de manière déterministe par Storjohann [114, Chap. 9].

1.8 Panorama des méthodes de résolution

En liaison avec les questions abordées dans les paragraphes précédents et plus particulièrement la résolution de système, le calcul du déterminant, de l'inverse et du polynôme caractéristique, une large gamme de méthodes directes a été développée. Les résultats donnés ou cités dans les chapitres suivants de ce mémoire s'appuient sur la plupart des techniques qu'elles font intervenir. Nous commentons cette gamme de méthodes très brièvement ici en citant les correspondances qui se présentent. Une compréhension d'ensemble du panorama échappe encore. Quelque peu arbitrairement, nous répertorions les méthodes en deux grandes classes : selon qu'elles développent un mécanisme d'élimination ou qu'elles font explicitement intervenir des puissances de matrices.

1.8.1 Éliminations

La résolution directe d'un problème d'algèbre linéaire qui met en jeu une matrice A passe fréquemment par une méthode d'élimination afin de transformer A en une matrice équivalente de forme "plus simple" permettant de résoudre le problème "plus facilement". Le type de l'élimination dépend de l'équivalence que l'on considère. En calcul formel trois types principaux d'éliminations sont concernés.

Si l'équivalence est à multiplication près, à droite ou à gauche, par des matrices de $\mathrm{GL}(n, \mathsf{K})$,

$$A' = LA \text{ ou } A' = LAR,\ A \in \mathsf{K}^{n\times n},\ L, R \in \mathrm{GL}(n, \mathsf{K}),$$

l'élimination est de type Gauss ou Gauss / Jordan [49, Chap. 2]. Les formes plus simples obtenues sont triangulaires ou diagonales, elles sont utilisées pour résoudre un système ou calculer le déterminant. En modèle algébrique on se reportera aux ouvrages [12, Chap. 2] ou [18, Chap. 16]. En modèle binaire sur $\mathbb{Z}$, la technique est combinée à la limitation de la taille des coefficients à la Bareiss [3] ou au théorème chinois [42, 19].

Si l'équivalence est à multiplication près, à droite ou à gauche, par des matrices unimodulaires sur un anneau principal R,

$$A' = UA \text{ ou } A' = UAV,\ A \in \mathsf{R}^{n\times n},\ U, V, U^{-1}, V^{-1} \in \mathsf{R}^{n\times n},$$

le principe de l'élimination de Gauss se conserve. La différence étant que les divisions utilisées pour la construction des matrices élémentaires sur un corps sont remplacées par des calculs de pgcd [86, 97]. Les formes plus simples calculées sont ici aussi triangulaires ou diagonales. On se référera à Storjohann pour un exposé des développements récents de ces méthodes pour le calcul des formes normales d'Hermite [60] et de Smith [111], ainsi que pour les liens calculatoires de cette approche avec l'élimination de Gauss [114, Chap. 5]. Ici c'est plutôt le modèle binaire qui est concerné (calculs de pgcd).

Si l'équivalence est la similitude de matrices sur un corps,

$$A' = P^{-1}AP,\ A \in \mathsf{K}^{n\times n},\ P \in \mathrm{GL}(n, \mathsf{K}),$$

l'élimination de Danilevski [30, 98] conduit en particulier à des formes Hessenberg creuses, comme dans [78] ou [52, Chap. 2], ou tridiagonales par blocs [114, Chap. 9]. Toujours en suivant le principe

de Gauss de faire apparaître des zéros de proche en proche dans la matrice, l'élimination effectue des opérations de lignes et celles correspondantes de colonnes pour la similitude. Comme vu au paragraphe précédent dans le cas d'une matrice $xI - A$, ou dans le cas de matrices polynomiales générales [126], l'équivalence par matrices unimodulaires peut être rapprochée de cette élimination par similitudes. En modèle algébrique les références précédemment citées s'intéressent au calcul du polynôme caractéristique (et donc du déterminant) et de la forme normale de Frobenius. L'approche se prolonge en modèle binaire pour des matrices entières [52, 57].

L'élimination de Danilevski est quasiment équivalente à l'approche de Krylov [81] (paragraphe qui suit) puisque les transformations correspondantes aux mises sous formes normales sont justement des matrices de type Krylov [49, Chap. 7, §5].

1.8.2 Puissances de matrices

La méthode de Le Verrier [85] calcule le polynôme caractéristique à partir des puissances de la matrice et des sommes de Newton [44, §47]. L'approche, redécouverte et améliorée [113, 43, 46], a été particulièrement utilisée en complexité parallèle (modèle algébrique) [29, 104, 48].

La méthode de Krylov / Lanczos [81, 84] [49, Chap. 7, §8] et ses nombreux développements dont nous détaillerons certaines variantes et applications dès le chapitre suivant, ne se fondent pas au départ sur un procédé d'élimination. Elle comporte surtout la construction d'un sous-espace de Krylov

$$\langle Y, AY, A^2Y, \ldots \rangle \subseteq \mathsf{K}^n, \text{ pour } Y \in \mathsf{K}^{n\times m} \text{ et } A \in \mathsf{K}^{n\times n}. \tag{1.14}$$

Cette approche a été mise au goût du jour en calcul formel sur des corps finis par Wiedemann [134] puis par Kaltofen, Pan et Saunders [72, 74]. Elle est essentiellement utilisée pour des calcul de polynômes minimaux qui sont des relations de dépendance dans des sous-espaces de type (1.14). Elle conduit de même au déterminant et au polynôme caractéristique (modèles algébrique et binaire) et a aussi été féconde en complexité parallèle [72, 73, 131].

Un dernier type d'approches se différencie puisqu'il permet d'obtenir des algorithmes de calcul du polynôme caractéristique χ_A sans effectuer de division. Berkowitz [11, 36] se base sur la formule de Samuelson [107] : si

$$A = \begin{bmatrix} a & R \\ S & \bar{A} \end{bmatrix} \in \mathsf{K}^{n\times n},$$

où $\bar{A}$ est $(n-1)\times(n-1)$, alors,

$$\chi_A(x) = (x-a)\chi_{\bar{A}}(x) - R(xI - \bar{A})^* S.$$

Cette identité fournit une récurrence pour calculer $\chi_{\bar{A}}(x)$ et le développement de la matrice adjointe $(xI - \bar{A})^*$ fait intervenir des puissances sous la forme $R\bar{A}^i S$. Chistov [22] produit le polynôme caractéristique comme polynôme miroir de

$$\det(I - xA) = (1/\det(I - xA))^{-1} \mod x^{n+1}.$$

Le développement de $1/\det(I - xA)$ conduit à calculer des éléments diagonaux de puissances de sous-matrices principales de A.

Des analogies précises entre l'approche de Le Verrier, celle de Samuelson / Berkowitz et celle de Chistov sont mises en évidence à l'aide d'interprétations combinatoires. Démarrées par Valiant avec l'algorithme de Berkowitz [122], ces études se poursuivent [88, 87, 112].

1.9 Résolution de systèmes linéaires

Pour une matrice $A \in \mathsf{K}^{n\times n}$ inversible et un vecteur b, résoudre le système linéaire $Ay = b$ est souvent plus simple que de résoudre les autres problèmes abordés. L'algorithme x-adique dans le cas polynomial ou p-adique sur $\mathbb{Z}$, initialement proposé par Moenck et Carter [91] et par Dixon [31], peut être accéléré pour donner la solution à un coût proche de celui du produit de matrices.

Théorème 1.9.1. *Pour $A \in \mathsf{K}[x]^{n\times n}$ non singulière de degré d et $b \in \mathsf{K}[x]^n$ de degré $O(nd)$, le vecteur $y \in \mathsf{K}(x)^n$ tel que $Ay = b$ peut être calculé en $\tilde{O}(n^\omega d)$ [116, Corollary 12] opérations dans K par un algorithme probabiliste Las Vegas.*

Même si des difficultés supplémentaires sont rencontrées [116, Conclusion], cette complexité devrait se généraliser au cas entier qui reste pour l'instant différent d'après le théorème suivant.

Théorème 1.9.2. *Pour $A \in \mathbb{Z}^{n\times n}$ non singulière et $b \in \mathbb{Z}^n$ tel que $\|b\| = O(\|A\|)$, le vecteur $y \in \mathbb{Q}^n$ tel que $Ay = b$ peut être calculé en $\tilde{O}\left(n^3 \log \|A\|\right)$ [91, 31] ou $\tilde{O}\left(n^{f(\omega)} \log \|A\|\right)$ avec $f(\omega) \approx 2.76$ [93, Lemma 5.7] opérations binaires par un algorithme probabiliste Las Vegas.*

Ces méthodes sont un analogue exact des méthodes numériques itératives avec correction du résidu. Pour une matrice $A \in \mathsf{K}[x]^{n\times n}$ de degré 1, l'algorithme x-adique se ramène à la matrice $B = I - xR = (A^{-1} \mod x)A$ et au système $By = c = (A^{-1} \mod x)b$. Vu la forme de B, la solution y est alors calculée à partir de $y(x) = B^{-1}(x)c = c + x(Rc) + x^2(R^2c) + \dots$ ce qui rappelle le principe de l'approche de Krylov. Il existe une version x-adique "par blocs" [116], nous l'interprétons ici pour rendre plus explicites ses liens avec les méthodes par blocs du chapitre suivant.

Pour $A \in \mathsf{K}[x]^{n\times n}$ et $B \in \mathsf{K}[x]^{n\times m}$, la solution à $AY = B$ est la fraction à gauche $Y = A^{-1}B$. On peut associer à Y une description à droite $Y = PQ^{-1}$ avec $P \in \mathsf{K}[x]^{n\times m}$ et $Q \in \mathsf{K}[x]^{m\times m}$. Pour Q donnée, on ne sait en général pas calculer cette dernière description, c'est-à-dire calculer P, avec la même complexité qu'au théorème 1.9.1 même si Q a un faible degré. La raison en étant que Y n'est pas forcément strictement propre. On peut en revanche calculer la description RQ^{-1} d'une autre fraction — elle strictement propre — qui conserve des invariants importants de Y (cf §2.1 et §5.3).

Proposition 1.9.3. *On suppose que $\det Q(0) \neq 0$. Pour $h > (n-1)d + \deg B$, soit $R \in \mathsf{K}[x]^{n\times m}$ définie par*

$$P(x) = \left(P(x)Q(x)^{-1} \bmod x^h\right) Q(x) + x^h R(x). \tag{1.15}$$

La fraction RQ^{-1} est strictement propre et le pgcd (à droite) de P et de Q est égal au pgcd de R et de Q. Pour A, B et Q données avec $m(\deg B)/d$ et $m(\deg Q)/d$ en $O(n)$, R peut être calculée en $\tilde{O}(n^\omega d)$ opérations dans K.

La démonstration sera trouvée en annexe page 51. L'identité (1.15) et le coût ont été donnés par Storjohann [116, §9]. À l'ordre des coefficients près, nous verrons cette identité comme une division euclidienne de polynômes matriciels. La fraction reste nous conduira à un nouvel algorithme de colonne réduction au paragraphe 5.3.

2. Approche Krylov / Lanczos

Nous formalisons ici l'approche de Krylov "par blocs" selon Wiedemann [134] et Coppersmith [26]. Très utilisée en analyse numérique sous une forme le plus souvent itérative, depuis le travail de Wiedemann [134] elle se révèle, sous une forme directe, aussi importante en calcul exact. Nous nous concentrons d'abord sur les grandes lignes de la méthode, c'est-à-dire sur les aspects communs à ses différentes applications telles que la résolution de systèmes linéaires, le calcul du déterminant ou du polynôme caractéristique. La plus grande partie de cette étude est tirée de nos travaux simultanés puis communs avec E. Kaltofen [67, 125, 127, 77]. Nous verrons plus loin (notamment aux chapitres 3 et 6) le détail des façons spécifiques dont l'approche est utilisée pour accélérer les résolutions des problèmes.

Pour $A \in \mathsf{K}^{n\times n}$ en entrée, le point de départ est de ramener le calcul du polynôme minimal de A, c'est-à-dire de la plus petite récurrence linéaire qui engendre la suite

$$A^i \in \mathsf{K}^{n\times n},\ i \geq 0,$$

au calcul du polynôme minimal d'une suite projetée

$$XA^iY \in \mathsf{K}^{l\times m},\ i \geq 0,$$

où $X \in \mathsf{K}^{l\times n}$ et $Y \in \mathsf{K}^{n\times m}$ sont des vecteurs ou des matrices. L'intérêt des projections est de permettre le calcul sur des dimensions plus petites que la dimension n initiale, donc de "condenser" l'information. Pour être sûr que cette information soit la même sur les deux suites (par exemple que leurs polynômes minimaux soient identiques) on introduit des choix aléatoires pour X et pour Y. C'est pourquoi l'approche donne essentiellement lieu à des algorithmes probabilistes.

Une première de nos contributions consiste à travailler avec des polynômes matriciels comme on sait le faire avec des polynômes scalaires. À ce titre, nous introduisons le polynôme minimal matriciel (définition 1.5.1) qui permet de cerner les invariants communs aux deux suites $(A^i)_{i\geq 0}$ et $(XA^iY)_{i\geq 0}$. C'est l'objet du paragraphe 2.1 avec un théorème de Cayley-Hamilton par blocs. Nous étudions ensuite au paragraphe 2.2 ce qui concerne le succès des choix aléatoires pour X et pour Y. L'analyse complète avait été menée par Wiedemann dans le cas de projections vectorielles. Dans le cas par blocs, manipuler le polynôme minimal matriciel nous a permis de faire aboutir l'analyse des probabilités de réussite entamée par Coppersmith puis Kaltofen. Le "tout matriciel" nous amène là à tirer parti d'outils tels que les fonctions arithmétiques de matrices.

Les applications de la méthode consistent à utiliser le polynôme minimal ainsi calculé pour résoudre d'autres problèmes (voir aussi le chapitre 6). Au paragraphe 2.3, vu comme une relation de dépendance entre vecteurs, le polynôme minimal permet la résolution de systèmes linéaires. Au paragraphe 2.4, égal au polynôme caractéristique d'une matrice obtenue par perturbation de A, il donne un moyen de calculer le déterminant.

Le polynôme minimal d'une suite récurrente est donc le cœur de la méthode de Krylov. Au paragraphe 2.5 nous introduisons pour son calcul un algorithme de type Knuth / Schönhage matriciel qui généralise les algorithmes les plus rapides du cas scalaire.

Nous concluons ce chapitre au paragraphe 2.6 avec de brefs commentaires sur l'approche très voisine de Lanczos.

2.1 Algorithme de Coppersmith / Wiedemann

L'objet de ce paragraphe, avec le théorème 2.1.1 ci-dessous, est d'expliciter les invariants communs aux deux suites $(A^i)_{i\geq 0}$ et $(XA^iY)_{i\geq 0}$. Pour une matrice $A \in \mathsf{K}^{n\times n}$ on considère donc l vecteurs lignes donnés par $X \in \mathsf{K}^{l\times n}$ et m vecteurs colonnes donnés par $Y \in \mathsf{K}^{n\times m}$. La méthode commence par calculer la suite

$$H_i = XA^iY \in \mathsf{K}^{l\times m}, i \geq 0. \tag{2.1}$$

Nous discuterons du choix de X et de Y ainsi que de l'ordre jusqu'auquel il suffit de calculer la suite aux paragraphes suivants. Le théorème de Cayley-Hamilton nous dit que cette suite est engendrée par le polynôme caractéristique $\chi_A(x)$ de A. D'après la proposition 1.5.2 on sait qu'elle est aussi engendrée matriciellement et on peut poursuivre avec le calcul de $F_X^{A,Y} \in \mathsf{K}[x]^{m\times m}$, son polynôme minimal (définition 1.5.1). Une version par blocs du théorème de Cayley-Hamilton montre alors que $F_X^{A,Y}$ conserve les facteurs invariants.

Théorème 2.1.1. *Le i-ème facteur invariant de $F_X^{A,Y}$ divise f_i le i-ème facteur invariant de A. De plus, il existe des matrices $W \in \mathsf{K}^{l\times n}$ et $Z \in \mathsf{K}^{n\times m}$ telles que pour tout i, $1 \leq i \leq \min\{l, m, \phi\}$, le i-ème facteur invariant de $F_W^{A,Z}$ est exactement f_i et telles que les $m - \min\{l, m, \phi\}$ facteurs restants soient égaux à 1. Dans ce cas, pour l et m fixés,*

$$\deg\det(F_W^{A,Z}(x)) = \deg f_1(x) + \cdots + \deg f_{\min\{l,m,\phi\}}(x) \leq n \tag{2.2}$$

est maximal [77, Theorem 1].

La quantité ϕ est le nombre de facteurs invariants de A (théorème 1.7.1). La démonstration donnée en annexe page 51 se ramène à étudier la fraction

$$X(xI - A)^{-1}Y = \sum_{i\geq 0} \frac{H_i}{x^{i+1}} = \sum_{i\geq 0} \frac{XA^iY}{x^{i+1}} \tag{2.3}$$

et à retrouver les facteurs invariants dans ses dénominateurs. Le rôle joué par cette fraction est précisé par le lemme suivant.

Lemme 2.1.2. *Une description $X(xI-A)^{-1}Y = N(x)D(x)^{-1}$ avec $D \in \mathsf{K}[x]^{m\times m}$ est irréductible si et seulement si $D = F_X^{A,Y}U$ pour une matrice U unimodulaire.*

Démonstration. Commençons par la condition nécessaire. Dans $X(xI - A)^{-1}Y$ les degrés des numérateurs sont strictement inférieurs aux degrés des dénominateurs. Chaque colonne de N a donc un degré strictement inférieur à la colonne correspondante de D. Par identification dans (2.3) on en déduit que chaque colonne de D engendre la suite des H_i de manière analogue à (1.9) et donc que D est un polynôme générateur. C'est donc un multiple du polynôme minimal, $D = F_X^{A,Y}V$ pour $V \in \mathsf{K}[x]^{m\times m}$, et on a aussi $N = N_1V$. Cela montre que si $V = U$ est unimodulaire alors la fraction est irréductible sinon le dénominateur réduit — diviseur de $F_X^{A,Y}U$ — ne pourrait être multiple de $F_X^{A,Y}$. La condition suffisante est l'unicité de la description irréductible donnée au théorème 1.6.5. □

Suivre la démarche globale de Krylov pour calculer des renseignements sur A c'est calculer la suite (2.1) puis calculer son polynôme générateur. Par exemple pour $l = m = 1$, le théorème 2.1.1 nous apprend que ce dernier sera probablement le polynôme minimal de A. Plus généralement on devrait avoir des informations sur les m premiers facteurs invariants de A.

2.2 Minorations des probabilités de réussite

Le principe de calculer la suite des H_i puis de calculer — au moins partiellement — son polynôme minimal, permet de résoudre différents problèmes. Les résolutions les plus efficaces sont obtenues à partir de choix aléatoires des projections X et Y avec des contraintes plus ou moins fortes en fonction du problème cible. À partir des travaux de Coppersmith [26], nous avons séparé l'étude des probabilités correspondantes en deux points [125, 127] :

- I) il faut d'abord que le choix de X et de Y permette de résoudre le problème, par exemple que le polynôme minimal de la suite des itérés de A et celui de la suite projetée coïncident ;
- II) il faut ensuite que le choix conduise à des coûts de résolution faibles, par exemple que le degré du polynôme minimal soit petit en fonction de m — typiquement de l'ordre de n/m.

Ce paragraphe donne une idée de la justification du fait que pour A quelconque, le polynôme minimal $F_X^{A,Y}$ a les facteurs invariants maximaux (théorème 2.1.1) et que son degré est au plus $\lceil n/m \rceil + \epsilon$ avec une bonne probabilité. Son degré est même $\lceil n/m \rceil$ pour un corps K assez grand.

Si l'on fixe A et Y, et que l'on note A_Y la restriction de A à l'espace de Krylov $\langle Y, AY, A^2Y \ldots \rangle$, le point I) va reposer sur la probabilité

$$\begin{aligned} \Phi_l(\phi) = &\quad \mathrm{Prob}_X \left[F_X^{A,Y} = F^{A,Y} \right] \\ = &\quad \mathrm{Prob}_{X'} \left[\dim\langle X', X'A_Y, X'A_Y^2, \ldots \rangle = \dim\langle Y, AY, A^2Y \ldots \rangle \right] \end{aligned} \tag{2.4}$$

où X' est une projection de X. Il y a un abus de notation dans la mesure où le premier espace est en lignes alors que le deuxième est en colonnes. Nous avons montré que cette probabilité dépend du nombre de facteurs invariants ϕ de A et qu'elle est assez grande pour assurer le succès des algorithmes [127]. L'étude se ramène à celles de Wiedemann [134, §VI] et de Kaltofen et Saunders [74, §2] pour $l = m = 1$. Plus précisément, pour des vecteurs u et v, Wiedemann s'est intéressé à la probabilité

$$\Phi_1(1) = \mathrm{Prob}_u \left[F_u^{A,v} = F^{A,v} \right]. \tag{2.5}$$

Il a montré qu'il existe une application linéaire ζ surjective de K^n vers l'ensemble des polynômes modulo $F^{A,v}$, telle que $F_u^{A,v} = F^{A,v}$ si et seulement si $\mathrm{pgcd}(\zeta(u), F^{A,v}) = 1$. La fonction ϕ d'Euler pour les polynômes sur K permet alors d'évaluer (2.5). Nous avons étendu ce résultat au cas matriciel. On peut voir que les évaluations de Φ correspondent à des évaluations de la fonction ϕ d'Euler pour des polynômes matriciels (au sujet des fonctions arithmétiques de matrices on peut consulter [23] et [96]).

Proposition 2.2.1. *Soit $\mathcal{M}_{A,Y}$ l'ensemble des matrices $l \times m$ sur $\mathsf{K}[x]$ dont les colonnes sont de degrés strictement inférieurs aux degrés des colonnes correspondantes de $F^{A,Y}$. Il existe une application linéaire surjective*

$$\zeta : \begin{cases} \mathsf{K}^{l \times n} \to \mathcal{M}_{A,Y} \\ X \mapsto N(x) \end{cases}$$

telle que $F_X^{A,Y} = F^{A,Y}$ si et seulement si les matrices $F^{A,Y}(x) \in \mathsf{K}[x]^{m \times m}$ et $\zeta(X) = N(x) \in \mathsf{K}[x]^{l \times m}$ sont premières entre elles (à droite). La probabilité $\Phi_l(\phi)$ est donc la probabilité pour qu'une matrice aléatoire de $\mathcal{M}_{A,Y}$ soit première avec $F^{A,Y}$.

On trouvera la démonstration de cette proposition en page 54. Les bornes inférieures que nous avons proposées pour Φ dans [127] ont été raffinées par Brent *et al.* [14]. Ces auteurs proposent notamment une expression exacte de la fonction.

Le deuxième point, II), pour le choix de X et de Y fait référence au degré du polynôme minimal, quantité qui gouverne le coût de son calcul (voir le théorème 1.5.3). On sait que ce degré est relié à la dimension d'un sous-espace de Krylov "tronqué".

Lemme 2.2.2. *Si δ est le premier entier tel que*

$$\dim\langle Y, AY, \ldots, A^{\delta}Y\rangle = \dim\langle Y, AY, \ldots\rangle = \nu$$

alors le polynôme minimal $F^{A,Y}$ de la suite A^iY est de degré exactement δ (au moins une de ses colonnes a un coefficient de degré δ) [125, Lemma 4.3].

Ce que l'on espère c'est que le degré du polynôme minimal soit de l'ordre de — voire soit exactement — $\lceil \nu/m \rceil$, ce qui impliquera que l'on ira le moins loin possible dans le calcul de la suite $(XA^iY)_{i\geq 0}$. La probabilité à laquelle on s'intéresse alors est :

$$\begin{aligned} \Theta_l(\phi, \lceil \nu/m \rceil, \epsilon) = &\quad \mathrm{Prob}_Y\left[\deg F^{A,Y} \leq \lceil \nu/m \rceil + \epsilon\right] \\ = &\quad \mathrm{Prob}_Y\left[\dim\langle Y, AY, \ldots, A^{\delta}Y\rangle \geq \dim\langle Y, AY, \ldots\rangle - \epsilon\right]. \end{aligned} \tag{2.6}$$

De même que pour Φ, nous avons proposé une minoration de Θ et montré le succès des algorithmes probabilistes correspondants [127]. On retrouve le fait courant que plus le corps K a beaucoup d'éléments, plus le succès est garanti. Les minorations sont évaluées directement par comptage même sur des corps de petite cardinalité. C'est ce qui introduit éventuellement dans (2.6) un petit décalage ϵ — sans conséquence en pratique — entre le degré atteint et le degré "idéal". Sur un corps K de cardinal élevé par rapport à n, les minorations peuvent se calculer à partir de projections X et Y génériques et en appliquant le lemme d'évaluation d'un polynôme en un point aléatoire (DeMillo/Lipton, Zippel, Schwartz) [51, Lemma 6.44].

2.3 Application au calcul d'un vecteur du noyau

Si A est singulière, le calcul du polynôme minimal permet de calculer directement des vecteurs de son noyau [26, 67]. Pour X et Y aléatoires, notons $f(x) = f_0 + xf_1 + \ldots x^d f_d \in \mathsf{K}[x]^m$ une des colonnes de $F_X^{A,AY}$. Par définition du polynôme générateur, on a

$$XAYf_0 + XA^2Yf_1 + \ldots + XA^{d+1}Yf_d = 0 \in \mathsf{K}^n. \tag{2.7}$$

En utilisant les propriétés assurées par Φ et Θ on montre que l'identité précédente reste vraie (avec une bonne probabilité) indépendamment de X :

$$AYf_0 + A^2Yf_1 + \ldots + A^{d+1}Yf_d = 0 \in \mathsf{K}^n. \tag{2.8}$$

Si f_r est le premier coefficient non nul de f, en mettant A en facteur on voit donc que l'on peut considérer le vecteur

$$\bar{w} = Yf_r + AYf_{r+1} + \ldots + A^{d-r}Yf_d = 0 \in \mathsf{K}^n, \tag{2.9}$$

pour construire un vecteur du noyau. Par construction on aura en effet une certaine puissance ι telle que $A^{\iota}\bar{w} = 0$.

Pour passer de (2.7) à (2.8) la probabilité portait sur X. En fait, la matrice Y aussi est concernée puiqu'il faut s'assurer que le vecteur $\bar{w}$ soit non nul. Cette condition sur Y est qu'au moins une de ses colonnes doit avoir une composante spectrale dans le noyau. Ceci est assuré avec une probabilité supérieure à $1 - 1/\mathrm{card}(\ker A)$ [26].

On remarque qu'une seule colonne du polynôme minimal est utilisée pour calculer un vecteur du noyau. Puisque la somme des degrés des colonnes du polynôme minimal est inférieure à n (théorème 2.1.1), il existe toujours une colonne de degré moins que n/m. Quand il s'agit de calculer le polynôme minimal avec en vue un vecteur du noyau, on peut donc toujours prendre $d \leq n/m$ (notamment au théorème 1.5.3) quitte à ne calculer ce polynôme minimal que partiellement.

La résolution d'un système linéaire non homogène impose des contraintes un peu plus fortes qui seront abordées au paragraphe 6.3.

2.4 Application au calcul du déterminant

Comme la résolution de systèmes linéaires, le calcul du déterminant peut être réduit au calcul du polynôme minimal [134, §V]. Si A est inversible, on peut toujours la pré-conditionner (voir §6.2) et supposer que son polynôme minimal est égal à son polynôme caractéristique.

D'après le théorème 2.1.1, si X et Y sont choisies de manière à ce que

$$\deg(\det(F_X^{A,Y}(x))) = n \tag{2.10}$$

alors le polynôme caractéristique de A est nécessairement

$$\chi_A(x) = \det(F_X^{A,Y}(x)) \tag{2.11}$$

et le déterminant en découle (donné par le terme constant). La condition (2.10) est assurée si X et Y sont choisies aléatoirement [67, 127]. C'est la fonction Φ — utilisée maintenant à la fois pour X et pour Y — qui le régit. De son côté, la fonction Θ, pour X et pour Y aussi, permet d'assurer que le polynôme minimal ait un degré inférieur à n/m.

On peut établir un parallèle entre ce type de calcul et ce qui a été développé au paragraphe 1.9 concernant l'approche x-adique. Storjohann [116] utilise la proposition 1.9.3 itérativement pour calculer le déterminant d'une matrice polynomiale $A'(x)$. Le procédé calcule la description $R(x)Q(x)^{-1}$ associée à $A'(x)^{-1}B$ puis la réduit en une fraction irréductible $T(x)F(x)^{-1}$. L'information sur le déterminant est finalement déduite de la forme normale de Smith du dénominateur $F(x)$. Dans le cas $A'(x) = xI - A$, le théorème 2.1.1 dit en effet qu'une description irréductible de $(xI - A)^{-1}B$ a pour dénominateur $F^{A,B}(x)$. On sait donc effectivement que $\det F^{A,B}(x)$ est un facteur du déterminant de $A'(x) = xI - A$.

2.5 Calcul du polynôme minimal à la Knuth / Schönhage

Dans les cadres des applications de l'approche de Krylov qui nous intéressent (plus particulièrement au §3.5) on peut utiliser pour le calcul du polynôme minimal une meilleure complexité que celle donnée au théorème 1.5.3. On va voir en effet que la suite récurrente mise en jeu est assez générique pour partir d'un algorithme d'Euclide matriciel "naïf" et combiner le produit de matrices rapide à l'approche récursive de Knuth / Schönhage pour le calcul du pgcd [77]. Ceci est à rapprocher des travaux de Thomé [120]. Nous allons démontrer le théorème suivant.

Théorème 2.5.1. *Soit $A \in \mathsf{K}^{n\times n}$ et soit S un sous-ensemble fini de K. On suppose connu et on note $\nu \leq n$ le degré maximal possible en (2.2). Si les composantes de X et de Y sont choisies uniformément et indépendamment dans S, alors avec une probabilité au moins $1-n(\lceil n/m\rceil+1)/|\mathsf{S}|$ on peut calculer le polynôme minimal $F_X^{A,Y}$ (pour $l = m$) en $\tilde{O}(m^\omega d)$ opérations dans K. Le procédé est un algorithme d'Euclide matriciel. Étant données X et Y telles que la suite des restes soit normale, l'algorithme est déterministe de coût $\tilde{O}(m^\omega d)$.*

On peut se reporter au coût (3.5) en page 25 pour un exemple d'utilisation de ce théorème. La démonstration du résultat sera conclue seulement en fin de section, au §2.5.4. Nous commençons par exposer l'algorithme d'Euclide pour des polynômes matriciels dans $\mathsf{K}[x]^{m\times m}$ et son lien avec le polynôme minimal. L'algorithme est une transcription directe de celui du cas scalaire si les matrices coefficients de tête des polynômes impliqués sont inversibles. Cette dernière condition — qui définit la notion de suite de restes normale — permet en effet de réaliser les divisions euclidiennes à chacunes des étapes et de faire baisser le degré. Nous montrons ensuite que la condition est satisfaite à partir de la suite $(XA^iY)_{i\geq 0}$ si X et Y sont suffisamment génériques. À l'image des sous-résultants, la condition est lue dans une matrice Hankel par blocs. Le théorème est alors prouvé en utilisant la version rapide d'Euclide à la Knuth / Schönhage et en justifiant que pour X et Y aléatoires, la suite des restes est normale avec une bonne probabilité.

2.5.1 Polynôme minimal et algorithme d'Euclide

On suppose pour simplifier l'exposé que les dimensions des deux matrices de projections sont identiques, soit que $l = m$. Comme souvent utilisé dans le cas scalaire [32, 66], le polynôme minimal de la suite peut être calculé par l'algorithme d'Euclide généralisé. En effet, si $F(x) = F_dx^d + F_{d-1}x^{d-1} + \ldots F_1x + F_0 \in \mathsf{K}[x]^{m\times m}$ engendre la suite $(H_i)_{i\geq 0}$, on a d'après (1.9) :

$$\begin{bmatrix} H_0 & H_1 & \ldots & H_d \\ H_1 & H_2 & \ldots & H_{d+1} \\ \vdots & \vdots & \ddots & \vdots \\ H_{d-1} & H_{d+1} & \ldots & H_{2d-1} \end{bmatrix} \begin{bmatrix} F_0 \\ F_1 \\ \vdots \\ F_d \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}. \tag{2.12}$$

Soit alors $H \in \mathsf{K}[x]^{m\times m}$ définie par $H(x) = H_0x^{2d-1} + H_1x^{2d-2} + \ldots + H_{2d-1}$. Par identification des coefficients, l'équation(2.12) est équivalente à l'existence de deux matrices $S(x)$ et $R(x)$ de degrés inférieurs à $d-1$ dans $\mathsf{K}[x]^{m\times m}$ telles que

$$x^{2d}S(x) + H(x)F(x) = M(x). \tag{2.13}$$

Les matrices $x^{2d}I$ et $H(x)$ peuvent donc être prises comme les entrées de l'algorithme d'Euclide pour le calcul du cofacteur $F(x)$. Dans le cas scalaire, la suite des restes d'Euclide est dite normale quand à chaque étape le degré baisse de 1 exactement. Le théorème des sous-résultants caractérise que la suite est normale par le fait que les sous-résultants sont non nuls [16]. Dans le cas matriciel on peut d'une manière analogue identifier des suites normales pour le calcul du polynôme minimal et en tirer parti.

Soient $M(x) = M_{2d}x^{2d}+\ldots+M_1x+M_0$ et $N(x) = N_{2d-1}x^{2d-1}+\ldots+N_1x+N_0$ dans $\mathsf{K}[x]^{m\times m}$. Si N_{2d-1} est inversible on peut diviser M par N de façon naïve :

$$\begin{cases} M = NQ + R, \text{ avec } \deg Q = 1, \ \deg R \leq 2d-2, \\ Q = N_{2d-1}^{-1}\left(M_{2d}x + M_{2d-1} - N_{2d-2}N_{2d-1}^{-1}M_{2d}\right). \end{cases} \tag{2.14}$$

Si la nouvelle matrice de tête R est aussi inversible (coefficient de degré $2d-2$), le processus peut se poursuivre. On dit alors que la suite matricielle est "normale" si tous les restes ont des matrices de tête inversibles. Auquel cas on définit :

$$\begin{cases} M^{(-1)} = M, \ M^{(0)} = N \\ M^{(i)} = M^{(i-2)} - M^{(i-1)}Q^{(i)}, \ 1 \leq i \leq d \end{cases} \tag{2.15}$$

avec $\deg M^{(i)} = 2d-1-i$. Cette récurrence construit $S^{(i)}$ et $F^{(i)}$ dans $\mathsf{K}[x]^{m\times m}$ telles que

$$M^{(-1)}S^{(i)} + M^{(0)}F^{(i)} = M^{(i)},\ 1 \le i \le d, \tag{2.16}$$

avec $S^{(i)}$ de degré $i-1$ et $F^{(i)}$ de degré i. On pose aussi $S^{(-1)} = I$, $S^{(0)} = 0$, $F^{(-1)} = 0$ et $F^{(0)} = I$.

2.5.2 Condition pour la normalité de la suite des restes

Le théorème suivant (démontré p. 54) et son corollaire établissent comme annoncé que le choix $M^{(-1)} = x^{2d}I$ et $M^{(0)} = H$ conduit à un polynôme minimal $F = F^{(d)}$ pour la suite $(H_i)_{i\ge 0}$ (on comparera (2.16) à (2.13)), et relient la normalité de la suite des restes à des conditions de rang dans une matrice Hankel par blocs.

Théorème 2.5.2. *Soit $H(x) = H_0x^{2d-1} + H_1x^{2d-2} + \ldots + H_{2d-1} \in \mathsf{K}^{m\times m}$ et $\nu = dm$. Si pour $1 \le k \le d$, les mineurs principaux $km \times km$ de la matrice Hankel par blocs*

$$\mathcal{M} = \begin{bmatrix} H_0 & H_1 & \ldots & H_{d-1} \\ H_1 & H_2 & \ldots & H_d \\ \vdots & \vdots & \ddots & \vdots \\ H_{d-1} & H_{d+1} & \ldots & H_{2d-2} \end{bmatrix} \in \mathsf{K}^{nu\times\nu} \tag{2.17}$$

sont non nuls alors avec $M^{(-1)} = x^{2d}I$ et $M^{(0)} = H$ on obtient :

i) *$M^{(i)}$ est de degré $2d-1-i$ $(0 \le i \le d)$ et sa matrice de tête $M^{(i)}_{2d-1-i}$ est inversible $(i \le d-1)$;*

ii) *$F^{(i)}$ est de degré i et sa matrice de tête $F^{(i)}_i$ est inversible $(0 \le i \le d)$;*

iii) *$S^{(i)}$ est de degré $i-1$ $(1 \le i \le d)$.*

De plus, l'algorithme d'Euclide produit un polynôme minimal $F^{(d)}$ pour la suite $(H_i)_{0\le i\le 2d-1}$.

Quand H est construite selon (2.1) le corollaire qui suit montre que $F^{(d)} = F_X^{A,Y}$. Pour la quantité ν qui est utilisée on peut aussi se reporter à (2.6).

Corollaire 2.5.3. *On note*

$$\nu = \deg(f_1(x)) + \cdots + \deg(f_{\min\{m,\phi\}}(x)) \le n.$$

où les f_i, $1 \le i \le \phi$, sont les facteurs invariants de A. On suppose que ν est divisible par m, soit $d = \nu/m$. Pour $H_i = XA^iY \in \mathsf{K}^{m\times m}$, $i \ge 0$, si $\mathcal{M}$ vérifie les hypothèses du théorème 2.5.2 alors $F^{(d)}$ est un polynôme minimal pour $(H_i)_{i\ge 0}$.

Démonstration. D'après le théorème 2.5.2, $F^{(d)}$ engendre $(H_i)_{0\le i\le 2d-1}$. Puisque Y a m colonnes,

$$\operatorname{rang}\,[Y \quad AY \quad A^2Y \quad \ldots] \le \nu,$$

donc pour la matrice Hankel par blocs infinie on a :

$$\operatorname{rang}\begin{bmatrix} H_0 & H_1 & \ldots \\ H_1 & H_2 & \ldots \\ \vdots & \vdots & \ddots \end{bmatrix} = \operatorname{rang}\,\mathcal{M} = \nu.$$

Ces deux matrices ont donc le même noyau et $F^{(d)}$ engendre $(H_i)_{i\ge 0}$. L'argument utilisé pour la minimalité de $F^{(d)}$ reste valable. □

Remarque 2.5.4. Nous n'avons donné l'algorithme d'Euclide que dans le cas où les matrices de tête restent inversibles jusqu'au bout, d'où l'hypothèse de divisibilité de ν par m. Le cas général demanderait de particulariser la dernière étape de division avec une matrice de rang $\nu \bmod m$ seulement.

2.5.3 Normalité de la suite des restes

Étant fixée $A \in \mathsf{K}^{n\times n}$, la normalité de la suite des restes associée et donc du polynôme minimal provient essentiellement de la généricité des projections X et Y. Ceci peut être partiellement déduit de [41, Lemma 4.1] dans le cas scalaire avec l'algorithme de Lanczos et de [67, Proposition 3] et [127, Proposition 2] dans le cas par blocs.

Soient $\mathcal{X}$ et $\mathcal{Y}$ des matrices $m \times n$ et $n \times m$ dont les coefficients sont des indéterminées $\xi_{j,k}$ et $\upsilon_{k,l}$ pour $1 \le k, l \le m$ et $1 \le k \le n$.

Lemme 2.5.5. *En gardant la notation*

$$\nu = \deg(f_1(x)) + \cdots + \deg(f_{\min\{m,\phi\}}(x)) \le n,$$

soit

$$\mathcal{M}(A,\mathcal{X},\mathcal{Y}) = \begin{bmatrix} \mathcal{H}_0 & \mathcal{H}_1 & \dots & \mathcal{H}_{d-1} \\ \mathcal{H}_1 & \mathcal{H}_2 & \dots & \mathcal{H}_d \\ \vdots & \vdots & \ddots & \vdots \\ \mathcal{H}_{d-1} & \mathcal{H}_{d+1} & \dots & \mathcal{H}_{2d-2} \end{bmatrix} \tag{2.18}$$

où $d = \lceil \nu/m \rceil$ et $\mathcal{H}_i = \mathcal{X}A^i\mathcal{Y}, i \ge 0$. La matrice $\mathcal{M}(A,\mathcal{X},\mathcal{Y})$ est de rang ν et ses ν sous-matrices principales $i \times i$, $\mathcal{M}_i(A,\mathcal{X},\mathcal{Y})$, sont inversibles.

La preuve sera trouvée p. 55. Le lemme implique en particulier les hypothèses du théorème 2.5.2 pour X et Y génériques et donc pour X et Y choisies aléatoirement.

2.5.4 Algorithme d'Euclide matriciel à la Knuth / Schönhage

On en arrive à la preuve proprement dite du théorème 2.5.1. En ré-écrivant (2.16), soit $U^{(i)} \in \mathsf{K}[x]^{(2m)\times(2m)}$ donnée par :

$$\begin{bmatrix} M^{(-1)} & M^{(0)} \end{bmatrix} \cdot U^{(i)} = \begin{bmatrix} M^{(i-1)} & M^{(i)} \end{bmatrix},\ 1 \le i \le d.$$

L'approche récursive de Knuth / Schönhage [79, 109] pour le calcul du pgcd de polynômes scalaires s'applique ici sans difficulté dans le cas matriciel. Nous ne reprendrons donc pas son exposé en détail (voir aussi [51, §11.1]). La matrice $U^{(i)}$ ne dépend que des $2i$ matrices coefficients de plus hauts degrés de $M^{(-1)}$ et de $M^{(0)}$, elle est de i (degré de $F^{(i)}$). Au niveau j de l'arbre des appels, $d/2^{j+1}$ appels récursifs ont lieu pour calculer $d/2^j$ matrices de degrés 2^j analogues à $U^{(2^j)}$. Le coût $C(j)$ d'un appel est $2C(j/2) + O(\mathrm{M}(m, 2^j))$ où $\mathrm{M}(m,k)$ est le coût du produit de deux polynômes matriciels de dimension m et de degré k. On sait que l'on peut prendre $\mathrm{M}(m,k) = O(m^\omega k \log k + m^2 k \log k \log\log k)$ [20]. Le coût global est donc en $\tilde{O}(m^\omega d)$. Avec les résultats de généricité du paragraphe précédent, cela donne finalement la démonstration suivante concernant le calcul du polynôme minimal.

Démonstration du théorème 2.5.1 page 17. On considère $\mathcal{X}$ et $\mathcal{Y}$ comme au lemme 2.5.5. Le polynôme $\prod_{i=1}^{d} \det(\mathcal{M}_{km}(A,\mathcal{X},\mathcal{Y}))$ est non identiquement nul de degré $md(d+1) \le n(\lceil n/m \rceil + 1)$ dans $\mathsf{K}[\dots, \xi_{j,k}, \dots, \upsilon_{k,l}, \dots]$. Il reste non nul en le point d'évaluation donné par X et Y avec la probabilité annoncée [51, Lemma 6.44], c'est-à-dire que les hypothèses du théorème 2.5.2 sont vérifiées et que l'algorithme récursif peut être appliqué. Le cas où m n'est pas un diviseur de ν est traité comme expliqué à la remarque 2.5.4. □

La généricité de la matrice Hankel (2.18) et du comportement de l'algorithme ne dépendent ici que de la généricité des projections X et Y. Ce fait a été remarqué au départ par Eberly [39] dans un contexte quelque peu différent. La généricité des projections assure en effet qu'il n'y a pas de *breakdown* (division par zéro) pendant l'algorithme de Lanczos (voir par exemple [83] au sujet de cette notion). Une preuve dans le cas scalaire mais avec pré-conditionnement de la matrice est donnée dans [41, Lemma 4.1]. Plus généralement les mineurs non nuls dans la matrice de Hankel traduisent l'absence de cas dégénérés pendant le calcul du polynôme minimal. Cela s'interprète aussi sous l'approche Berlekamp / Massey où alors tous les résidus sont non nuls à chaque étape. C'est d'ailleurs ce dernier aspect qui permet d'implanter la stratégie de "terminaison anticipée" proposée par Lobo. Le polynôme minimal est produit dès qu'une quantité s'annule (*e.g.* le résidu avec Berlekamp / Massey ou un mineur dans la matrice de Hankel) ou plutôt, heuristiquement, reste nulle pendant quelques étapes (voir [34, §3] et [33, §6.3]).

2.6 Krylov versus Lanczos

Les paragraphes précédents ont adopté le point de vue Krylov / Wiedemann. Il y a eu distinction entre deux grandes phases : le calcul de la suite H_i et le calcul du polynôme minimal (puis son utilisation par exemple pour le déterminant). Dans l'approche Lanczos ces deux phases s'entremêlent en une seule itération. Dans le même temps que les vecteurs de l'espace de Krylov, les coefficients du polynôme minimal sont mis à jour par produits scalaires et/ou des vecteurs du type (2.9) sont calculés à la volée par exemple pour la résolution d'un système linéaire. Lambert a proposé — sur des corps finis — une étude unifiée des deux techniques [83] (voir aussi [25, 41] ou [33, §6.4]). Nous verrons une comparaison succincte des coûts des deux approches au paragraphe 6.1. Elles ne présentent pas de différences essentielles au plan théorique.

3. Calculs sans division

Nous nous intéressons au calcul sans division du déterminant d'une matrice. Pour A une matrice $n \times n$ sur un anneau R, il s'agit de calculer $\det A$ en n'effectuant que des opérations dans l'anneau. Ce sujet, d'intérêt théorique, a des implications pratiques puisqu'il a aussi conduit jusqu'à présent à de bons algorithmes par exemple sur $\mathbb{Z}$ ou sur des anneaux de polynômes à plusieurs variables (on pourra par exemple consulter les expérimentations de [2]).

La technique d'élimination des divisions de Strassen (voir ci-dessous) appliquée à l'elimination de Gauss ainsi que les algorithmes de Berkowitz et de Chistov (brièvement abordés au §1.8.2), conduisent à une complexité de $O\left(n^{\omega+1}\right)$ opérations dans R. Cela correspond approximativement à n fois le coût algébrique avec divisions.

Kaltofen a le premier montré que ce facteur pouvait être réduit. Nous exposons sa méthode initiale au paragraphe 3.1 et montrons au paragraphe suivant comment nous l'avons conjointement améliorée. Il est un fait que la méthode que nous proposons, comme celles de Strassen, Berkowitz et Chistov (nous empruntons là une remarque de [2]), calcule le polynôme caractéristique pour obtenir le déterminant par voie de conséquence. Nous le préciserons au dernier paragraphe.

3.1 Élimination des divisions et algorithme de Kaltofen

Strassen a donné une technique pour éliminer les divisions d'un programme sur un corps K qui calcule un polynôme [119]. Supposons donné un tel programme pour calculer le déterminant qui est un polynôme de degré n en les entrées de la matrice. La méthode applique une homotopie en utilisant un point où le programme n'effectue pas de division (ou des divisions par 1) et exécute le programme en plongeant les opérations dans l'anneau des séries formelles modulo z^{n+1}. Grâce à l'homotopie, les seules divisions sont des divisions par des séries du type $1 - za$ et peuvent être remplacées par des multiplications puisque $1/(1 - za) \equiv 1 + az + a^2z^2 + \ldots + a^nz^n \mod z^{n+1}$.

Plus précisément, pour $A \in \mathsf{R}^{n\times n}$ en entrée, Kaltofen utilise l'homotopie $(1 - z)C + zA$ avec C une matrice bien choisie. Il suffit alors de calculer

$$\det((1 - z)C + zA) \mod z^{n+1}$$

puisque de là :

$$\det A = \det((1 - z)C + zA)_{|z=1} = (\det((1 - z)C + zA) \mod z^{n+1})_{|z=1}. \tag{3.1}$$

Le programme algébrique sous-jacent est l'algorithme de Wiedemann c'est-à-dire l'algorithme du paragraphe 2.4 avec $l = m = 1$. Le polynôme minimal de la suite est calculé par l'algorithme d'Euclide du paragraphe 2.5 dans le cas de polynômes scalaires. Le choix de la matrice C est fixé comme suit.

Lemme 3.1.1. *On peut construire sans division une matrice C compagnon $n \times n$ et un vecteur $v \in \mathsf{K}^n$ tels que l'algorithme d'Euclide appliqué au calcul du polynôme minimal π_C de la suite $(e_1^T C^i v)_{i\geq 0}$ n'effectue pas de division. Le vecteur e_1 est le premier vecteur canonique. Le polynôme $\pi_C \in \mathsf{K}[x]$ est de degré n exactement [66, §2].*

Pour un degré n, le coût algébrique de l'algorithme de Wiedemann est celui du calcul de $2n$ termes d'une suite (2.1) plus le coût du calcul de son polynôme minimal (théorème 1.5.3), ce qui donne au total $\tilde{O}(n^\omega)$ opérations dans K pour calculer le déterminant. Le programme sans division est l'algorithme de Wiedemann pour la matrice $\tilde{A}(z) = (1-z)C + zA$ en utilisant les projections e_1 et v du lemme. La manipulation des séries formelles induit un surcoût par rapport au n^ω algébrique. La technique des pas de bébé / pas de géant judicieusement appliquée au calcul de

$$h_i(z) = e_1^T \tilde{A}(z)^i v = e_1^T((1-z)C + zA)^i v,\ i = 0, \ldots, 2n-1 \tag{3.2}$$

restreint le surcoût à un facteur moins que n. La méthode permet de calculer le déterminant en $\tilde{O}\left(n^{3+1/2}\right)$ opérations dans R sans produit rapide de matrices ou en $\tilde{O}\left(n^{g(\omega)}\right)$, pour $g(\omega) \approx 3.03$, avec produit rapide de matrices [66].

3.2 Calcul du déterminant sans division

Via la technique de Strassen, réduire le coût du calcul sans division revient à réduire le surcoût lié à la manipulation des séries formelles. Au paragraphe précédent ce surcoût vient essentiellement du calcul (3.2) qui fait intervenir jusqu'à la $(2n-1)$-ème puissance de $\tilde{A}(z)$. Basés sur le paragraphe 2.4, nous avons remarqué qu'une version par blocs de tailles m avec un polynôme minimal de degré $d = n/m$ (on supposera que m divise n exactement) n'allait conduire qu'à une puissance $2d-1 = 2n/m - 1$ au plus et donc réduire la complexité [77].

Pour un point normal où l'algorithme par blocs sur K n'effectue pas de division, avec les données C et v du lemme 3.1.1 en dimension d, on peut prendre

$$C_m = \operatorname{diag}(C, C, \ldots, C) \in \mathsf{K}^{n \times n}$$

et les projections à gauche et à droite

$$E = \begin{bmatrix} e_1^T \\ e_{d+1}^T \\ \vdots \\ e_{n-d+1}^T \end{bmatrix} \in \mathsf{K}^{m \times n}, \qquad V = \begin{bmatrix} v & 0 & \ldots & 0 \\ 0 & v & \vdots & \vdots \\ \vdots & 0 & \vdots & \vdots \\ \vdots & \vdots & \vdots & 0 \\ 0 & 0 & \ldots & v \end{bmatrix} \in \mathsf{K}^{n \times m}.$$

où les e_i sont les vecteurs canoniques. On vérifie que toutes les colonnes du polynôme minimal de la suite $EC_m^i V$, $i \geq 0$, sont de degré $d = n/m$. C'est aussi dire, en particulier, que le déterminant de ce polynôme minimal est égal au polynôme caractéristique de C_m (théorème 2.1.1). Au contraire de la méthode générale du paragraphe 2.4, le choix déterministe de E, C et V ne demande pas de conditionnement en entrée. De même qu'au lemme 3.1.1 du cas scalaire, l'algorithme d'Euclide matriciel du paragraphe 2.5 appliqué à la suite $EC_m^i V$ s'effectue sans division. Plus précisément, en reprenant la construction (2.16), la matrice de tête de $M^{(i)}$ ainsi que la matrice de tête et celle de degré constant de $F^{(i)}$ sont au signe près la matrice identité, i allant jusqu'à l'indice d.

Nous donnons une version simplifiée de l'algorithme en page 25. La suite récurrente est calculée en généralisant les pas de bébé et les pas de géant vus plus haut suivant un paramètre r. La complexité est obtenue en optimisant le choix de ce paramètre et de la taille m des blocs. Cela fait intervenir le produit rapide de matrices rectangulaires.

Théorème 3.2.1. *Pour un nombre réel κ, $0 \leq \kappa \leq 1$, soit*

$$\omega(\kappa) = \begin{cases} 2 + o(1), & 0 \leq \kappa \leq \alpha = 0.2946, \\ (2(1-\kappa) + (\kappa - \alpha)\omega)/(1-\alpha), & \alpha < \kappa \leq 1. \end{cases}$$

Le produit d'une matrice $n \times n$ par une matrice $n \times n^\kappa$ peut être calculé en $\mathsf{MM}(n, n^\kappa) = O(n^{\omega(\kappa)})$ opérations dans R [27], [61, (8.1)].

Bien entendu, pour $\kappa = 1$ on retrouve l'exposant ω. Quitte à arrondir les valeurs ou à considérer des puissances exactes, on définit γ, s et β par

$$m = 2n^\gamma, \quad rs = 2n/m = 2d, \quad r = n^{1-\gamma-\beta}, \quad s = n^\beta.$$

Algorithme Déterminant
Entrée : $A \in \mathsf{R}^{n \times n}$
Sortie : $\det A$

$\tilde{A}(z) = (1-z)C_m + zA$

/* *Calcul des* $H_i(z) = E\tilde{A}(z)^i V$, $0 \leq i \leq 2d-1$ */
1.1 $W_j(z) := \tilde{A}(z)^j V$, $0 \leq j \leq r-1$ /* *Le bébé et ses blocs* */
1.2 $Z(z) := \tilde{A}(z)^r$
1.3 $U_k(z) := EZ(z)^k$, $0 \leq k \leq s-1$ /* *Le géant et ses blocs* */
1.4 $H_{kr+j}(z) := U_k(z)W_j(z)$, $0 \leq j \leq r-1$, $0 \leq k \leq s-1$

/* *Calcul du polynôme minimal* $F(x,z) = x^d + F_{d-1}(z)x^{d-1} + \ldots + F_0(z)$ */
$F(x,z) :=$Polynôme minimal $(H_i(z),\ 0 \leq i \leq 2d-1)$
$\det A := \det F_0(1)$.

Puisque l'étape 1.3 calcule le même type de quantités que l'étape 1.1 mais sur des polynômes de plus hauts degrés, le meilleur compromis conduira à $r \geq s$ (i.e. $\beta \leq (1-\gamma)/2$). Les coûts des étapes 1.1, 1.2 et 1.4 sont dominés par

$$\tilde{O}(rn^\omega). \tag{3.3}$$

Pour l'étape 1.3 on peut montrer que le produit de matrices rectangulaires permet d'avoir

$$\tilde{O}\left(\frac{n}{m}\mathsf{MM}(n, ms)\right). \tag{3.4}$$

Pour ce qu'il reste à calculer, le choix de E, C_m et V fait double-emploi. On a vu qu'il assure d'une part que l'algorithme n'a pas besoin de diviser. Il conduit d'autre part à une suite de restes normale dans l'algorithme d'Euclide matriciel. Pour le calcul du polynôme minimal $F(x,z)$ modulo z^{n+1} le théorème 2.5.1 donne donc $(m^\omega d) \times n$, soit :

$$\tilde{O}\left(m^{\omega-1}n^2\right). \tag{3.5}$$

L'algorithme se conclut par le calcul du déterminant de la matrice $F_0(1)$, $m \times m$, ce qui peut se faire en $\tilde{O}\left(m^{\omega+1}\right)$, donc en un coût qui n'est pas dominant.

Le coût final est obtenu en minimisant l'exposant de n dans le maximum de (3.3), (3.4) et (3.5). Quand la taille de bloc m augmente, c'est-à-dire que l'exposant γ augmente, alors les coûts (3.3)

et (3.4) s'accroissent alors que (3.5) décroît. Quand le nombre s (exposant β) de pas de géant augmente, (3.3) diminue, (3.4) augmente aussi et (3.5) reste constant. La valeur

$$\beta_{\text{opt}} = \frac{2 - 2\,\gamma + \omega\,\gamma - \omega}{1 - \omega + \alpha}$$

égalise les coûts (3.3) et (3.4), et la valeur

$$\gamma_{\text{opt}} = \frac{-3 + 3\,\omega - \alpha - \omega^2 + \omega\,\alpha}{-2 + 2\,\omega - \omega^2 + \omega\,\alpha}$$

minimise le maximum des trois coûts et donne donc l'optimum. Finalement, selon le type de l'arithmétique utilisée, on arrive aux coûts suivants.

Théorème 3.2.2. *Le déterminant de $A \in \mathsf{R}^{n\times n}$ peut être calculé sans division en temps $\mathcal{D}et_{\mathsf{R}}(n) = O(n^{2.7})$. Le coût est $\tilde{O}\left(n^{3+1/5}\right)$ sans produit rapide de matrices et $O(n^{3+1/3})$ si la transformée de Fourier rapide n'est pas non plus utilisée [77].*

3.3 Polynôme caractéristique et matrice adjointe

Plus que le déterminant, l'algorithme sans division a calculé $F(x,z)$ qui donne le polynôme caractéristique de $\tilde{A}$:

$$\chi_{\tilde{A}}(x,z) = \det F(x,z) \mod z^{n+1}.$$

Le polynôme caractéristique χ_A de A peut donc être calculé comme $\det F(x,1)$. Pour obtenir cette quantité sans division on applique le nouvel algorithme sans division à $A = F(x,1) \in \mathsf{R}[x]^{m\times m}$ de degré n/m. L'algorithme est exécuté par exemple modulo x^{n+1} et puisque le produit de polynôme sans division est en $O(n \log n \log\log n)$ [20], à partir de $F(x,1)$, $\chi_A(x)$ est obtenu en $\tilde{O}\,(n\mathcal{D}et_{\mathsf{R}}(m))$. Comparons cette complexité à $\mathcal{D}et_{\mathsf{R}}(n)$. En prenant (3.5) comme expression de $\mathcal{D}et_{\mathsf{R}}(n)$, l'exposant

$$1 + \gamma(2 + \gamma(\omega - 1)) = 1 + 2\gamma + (\omega - 1)\gamma^2,$$

de n dans $n\mathcal{D}et_{\mathsf{R}}(m)$ est plus petit que l'exposant

$$2 + (\omega - 1)\gamma$$

de n dans $\mathcal{D}et_{\mathsf{R}}(n)$ si

$$(\omega - 1)\gamma^2 + (3 - \omega)\gamma - 1 \leq 0.$$

Cette inégalité est effectivement vérifiée pour $\omega \geq 2$, la valeur de α donnée au théorème 3.2.1 et γ_{opt}. Outre cette remarque, par différentiation automatique en mode inverse la complexité du déterminant est valable pour le calcul de l'adjointe [4].

Corollaire 3.3.1. *Les exposants de n au théorème 3.2.2 sont valables pour le calcul du polynôme caractéristique et de l'adjointe sur R.*

4. Matrices sur $\mathbb{Z}$

Comme nous l'avons évoqué dans l'introduction, le coût d'un calcul sur une matrice entière, en termes d'opérations sur les bits, est *a priori* relié à la longueur binaire des entiers que ce calcul amène à manipuler.

On l'a vu en (1.5), le déterminant peut avoir jusqu'à $\tilde{O}(n \log \|A\|)$ bits. Quant à son calcul, en appliquant le théorème chinois on peut schématiquement borner le coût binaire suivant

$$\text{coût binaire} \ \leq \ \text{coût arithmétique} \times \ \text{taille maximum en sortie.} \tag{4.1}$$

Pendant longtemps cela a conduit à considérer que ce coût binaire était de $\tilde{O}\left(n^{\omega+1} \log \|A\|\right)$. Nous proposons un survol du sujet dans [76]. C'est en introduisant une technique à base de perturbations de rang k, d'abord développée pour des matrices creuses [130], que nous avons montré comment passer en deçà du seuil $n^\omega \times n$ (ou $n^3 \times n$). Le paragraphe 4.1 présente la méthode. Il s'est ensuite avéré que la technique développée pour le calcul sans division pouvait aussi servir en complexité binaire et même améliorer les coûts [69], nous le verrons au paragraphe 4.2. On terminera avec quelques prolongements en particulier pour le calcul du polynôme caractéristique.

Une remarque à partir de ces améliorations du calcul du déterminant est que les nouveaux algorithmes calculent tous — explicitement ou implicitement — la forme normale de Smith de la matrice[1]. Le déterminant s'en déduit d'après (1.6) comme produit des facteurs invariants. En l'état des connaissances la forme normale n'est pas vraiment "plus difficile à calculer".

4.1 Le déterminant par perturbations de rang k

Puisque résoudre un système linéaire est *a priori* plus simple que de calculer le déterminant (§1.9), il est naturel d'utiliser la résolution de ce premier problème comme brique de base. Cette idée a été proposée par Pan dans [99, Appendix] et [100] pour calculer le déterminant d'une matrice générique A à partir de systèmes $Ax = b$ avec seconds membres aléatoires. On peut notamment montrer que deux résolutions conduisent au dernier facteur invariant.

Proposition 4.1.1. *Soient $A \in \mathbb{Z}^{n\times n}$ une matrice inversible et b_1, b_2 deux vecteurs aléatoires aux composantes entre 0 et $6 + 2n(\log n + \log \|A\|)$. En supposant que les fractions rationnelles sont réduites, soit y_1 et y_2 les solutions à $Ay_1 = b_1$ et $Ay_2 = b_2$. Alors si σ_1 est le ppcm des dénominateurs de y_1 et de y_2 on a*

$$\mathrm{Prob}_{b_1,b_2}[\sigma_1 = s_1] > 1/3$$

où s_1 est le plus grand facteur invariant de A [1, 40].

Les coûts de résolution d'un système donnés par les théorèmes 1.9.1 et 1.9.2 sont donc valables asymptotiquement pour le calcul du plus grand facteur invariant. Ce que Pan utilisait est le fait que génériquement et même en moyenne, le plus grand facteur invariant est égal au déterminant (théorème 1.2.2).

[1] C'est aussi vrai pour l'algorithme de Storjohann [116] prouvé pour des matrices polynomiales et dont la généralisation au cas entier est attendue (voir §2.4 et §5.4).

Corollaire 4.1.2. *Si $\mathcal{S}_{\|A\|}(n)$ est le coût binaire de résolution d'un système linéaire avec $\|b\| = O(n + \|A\|)$ alors le plus grand facteur invariant de A se calcule en $\tilde{O}\left(\mathcal{S}_{\|A\|}(n)\right)$ opérations binaires par un algorithme probabiliste de type Monte Carlo. Pour des matrices aléatoires telles que $\log \|A\| > 3 \log n$ il existe un algorithme probabiliste Monte Carlo dont le coût en moyenne est $\tilde{O}\left(\mathcal{S}_{\|A\|}(n)\right)$ [40, 76].*

On a vu et justifié sur l'exemple en page 2 que la résolution de systèmes ainsi utilisée ne peut conduire qu'à la connaissance du plus grand facteur invariant. Nous avons introduit des perturbations aléatoires de rang k comme moyen de trouver les autres facteurs invariants et de là la valeur du déterminant [130, 40]. Pour deux matrices $U \in \mathbb{Z}^{n \times k}$ et $V \in \mathbb{Z}^{k \times n}$ on considère la perturbation $A + UV$. Intuitivement, une telle perturbation de rang k ne peut perturber que les k plus grands facteurs invariants de A et elle les "détruit" tous si elle est suffisamment générique. Auquel cas le plus grand facteur invariant de $A + UV$ est $s_{k+1}p$, où p est un facteur premier avec le déterminant de A. Et donc le pgcd du dernier facteur invariant de A et de celui de $A + UV$ est égal à s_{k+1}.

Proposition 4.1.3. *On suppose $n \geq 6$. Il existe un entier β de taille en $\tilde{O}(\log n + \log\log \|A\|)$ et une constante κ tels que si $U_i \in \mathbb{Z}^{n \times k}$ et $V_i \in \mathbb{Z}^{k \times n}$, $1 \leq i \leq \kappa$, sont choisies aléatoirement avec leurs composantes entre 0 et β alors*

$$\mathrm{Prob}_{U_i, V_i, 1 \leq i \leq \kappa}[\mathrm{pgcd}(s_1, s_1^{(1)}, \ldots, s_1^{(\kappa)}) = s_{k+1}] > 1/2$$

où $s_1^{(i)}$ est le plus grand facteur invariant de $A + U_i V_i$ [40, §3.4].

En appliquant le corollaire 4.1.2 aux matrices $A + U_i V_i$ on en déduit donc que n'importe lequel des facteurs invariants peut aussi être calculé en temps $\tilde{O}\left(\mathcal{S}_{\|A\|}(n)\right)$. Pour les calculer tous, on a besoin d'une remarque supplémentaire. Puisque leur produit est le déterminant et puisqu'ils se divisent entre eux, on vérifie qu'il existe au plus $\mu = \sqrt{2 \log |\det A|} = \tilde{O}(\sqrt{n} + \log \|A\|)$ facteurs invariants distincts ([130, Theorem 3] et [40, Theorem 4.1]). On peut alors mettre en œuvre une recherche binaire pour les trouver. Le mécanisme de cette recherche est que si l'on calcule s_i et s_{i+k} et que l'on constate que $s_i = s_{i+k}$ alors, d'après la règle de divisibilité, on sait que $s_i = s_{i+1} = \ldots = s_{i+k-1} = s_{i+k}$. Sinon la recherche binaire peut continuer sur les deux intervalles moitiés. Ceci conduit au calcul de tous les facteurs invariants en appliquant de l'ordre de $\mu \log n$ fois le corollaire et la proposition précédents. En $\tilde{O}(\sqrt{n} + \log \|A\|)$ résolutions de systèmes linéaires la forme normale de Smith est connue. Ceci pouvant se compléter par un algorithme qui tire parti du produit rapide de matrices on obtient les complexités suivantes.

Théorème 4.1.4. *Il existe un algorithme probabiliste Monte Carlo qui calcule la forme normale de Smith et donc le déterminant de $A \in \mathbb{Z}^{n \times n}$ en $\tilde{O}\left(n^{3.5} \log^{2.5} \|A\|\right)$ ou $\tilde{O}\left(n^{2+\omega/2} \log^{1.5} \|A\|\right)$ opérations binaires [40].*

4.2 Le déterminant à partir de l'algorithme sans division

Les techniques des algorithmes sans division, c'est-à-dire sur un anneau R, conduisent souvent à de bons algorithmes quand on tient compte du coût des opérations dans R. En faisant le rapprochement entre l'exposant $2 + \omega/2$ du théorème 4.1.4 et son exposant — en complexité algébrique — de 1992 [66], Kaltofen a constaté que son algorithme sans division donnait lieu à un algorithme sur $\mathbb{Z}$ [69]. Nous allons voir qu'il en est de même pour l'algorithme par blocs du paragraphe 3.2.

Le gain obtenu en réduisant le surcoût lié à la manipulation des séries formelles se retrouve comme un gain sur la manipulation des entiers.

Il ne s'agit pas d'exécuter directement l'algorithme de la page 25 en prenant $\mathsf{R} = \mathbb{Z}$. Il s'agit d'appliquer à $A \in \mathbb{Z}^{n\times n}$ le traitement qui était appliqué à $\tilde{A}(z)$. Prendre directement $\mathsf{R} = \mathbb{Z}$ est peut-être réalisable, c'est une étude que nous avons commencée. Cela conduit à analyser les entiers qui apparaissent au cours des calculs. Ce n'est cependant pas nécessaire ici pour le calcul du déterminant. Sa taille bornée *a priori* permet de développer un algorithme à partir de l'algorithme algébrique en utilisant le théorème chinois.

Les phases de calcul sur A sont les mêmes que sur $\tilde{A}(z)$ page 25. En suivant la technique présentée au paragraphe 2.4, quitte aussi à pré-conditionner A pour bien aboutir au polynôme caractéristique, on choisit aléatoirement deux matrices $X \in \mathbb{Z}^{m\times n}$ et $Y \in \mathbb{Z}^{n\times m}$. Ce choix assure notamment que l'on peut appliquer l'algorithme d'Euclide matriciel pour calculer le polynôme minimal et que ce dernier est de degré $d = n/m$ (théorème 2.5.1). On suppose à nouveau pour simplifier que m divise n. Le calcul de la suite $H_i = XA^iY$ se décompose de la même manière qu'au paragraphe 3.2. Les coûts (3.3) et (3.4) restent valables avec la taille des entiers qui joue le même rôle que l'ordre auquel les séries étaient tronquées.

Comme expliqué plus haut, le polynôme minimal $F \in \mathbb{Q}[x]^{m\times m}$ n'est pas calculé explicitement. À partir de la donnée de la suite des H_i on termine en utilisant le théorème chinois. Le polynôme minimal puis le déterminant de son terme constant sont calculés modulo $\tilde{O}(n \log \|A\|)$ nombres premiers et $\det A$ en est déduit [77]. L'algorithme est probabiliste. Si la matrice de tête de F est inversible on sait que le déterminant de F est de degré n exactement. On est donc sûr d'avoir calculé le polynôme caractéristique et le déterminant. Cette dernière remarque permet de vérifier la sortie et de donner un algorithme Las Vegas.

Théorème 4.2.1. *Le déterminant de $A \in \mathbb{Z}^{n\times n}$ peut être calculé par un algorithme Las Vegas en $\tilde{O}\left(n^{2.7} \log \|A\|\right)$ opérations binaires. Le coût est $\tilde{O}\left(n^{3+1/5} \log \|A\|\right)$ sans produit rapide de matrices et $\tilde{O}\left(n^{3+1/3} \log \|A\|\right)$ sans non plus de transformée de Fourier rapide [77].*

Les exposants sont obtenus comme au paragraphe 3.2. Kaltofen et Pan ont étendu ces coûts en proposant des versions adaptatives de la méthode [70]. Un algorithme est dit adaptatif si son coût dépend des valeurs de ses sorties et pas seulement de bornes au pire sur ces dernières. Les coûts ne sont pas plus faibles dans le pire cas mais peuvent être bien meilleurs par exemple si le déterminant se révèle petit *a posteriori*.

4.3 Forme normale de Smith et polynôme caractéristique

Pour généraliser les complexités du théorème 4.2.1 à d'autres problèmes que le déterminant on dispose de peu d'outils de réduction comparé au contexte de la complexité algébrique du paragraphe 3.3. La forme normale de Smith et la forme normale de Frobenius sont cependant accessibles.

Le calcul de la forme normale de Smith découle directement du calcul du déterminant. En effet, l'algorithme du paragraphe précédent peut être étendu au calcul du polynôme caractéristique d'une matrice A' obtenue par pré-conditionnement de A (pré-multiplication par des matrices à droite et à gauche). Le surcoût est celui du calcul du déterminant de $F \in \mathsf{K}[x]^{m\times m}$ (non plus seulement du déterminant de son terme constant). Sur $\mathsf{K}[x]$ avec K un corps fini, d'après le théorème 5.2.1, le coût du calcul du déterminant est en $\tilde{O}(m^\omega d)$. Cela donne donc un surcoût de $\tilde{O}\left(m^{\omega-1}n^2 \log \|A\|\right)$, en utilisant $\tilde{O}(n \log \|A\|)$ nombres premiers, pour calculer le polynôme caractéristique $\chi_{A'} \in \mathbb{Z}[x]$ de A' à l'aide du théorème chinois. L'exposant global n'est pas augmenté puisque c'est de l'ordre de la quantité (3.5) déjà comptabilisée.

À partir de $\tilde{O}(\log n + \log\log \|A\|)$ tels calculs de polynômes caractéristiques, l'algorithme probabiliste Monte Carlo de Giesbrecht [55, Theorem 4.2] permet de calculer les facteurs invariants de A. La méthode procède par calculs de pgcd sur les coefficients de ces premiers. Ces coefficients sont en effet des sommes de déterminants extraits de dimensions données et contiennent l'information pour la forme de Smith.

Corollaire 4.3.1. *Les exposants du théorème 4.2.1 sont valables pour le calcul de la forme normale de Smith de $A \in \mathbb{Z}^{n\times n}$ par un algorithme probabiliste Monte Carlo.*

Sans préconditionnement de la matrice A en entrée, d'après le théorème 2.1.1, l'algorithme va calculer

$$\chi'_A(x) = \prod_{i=1}^{\min\{m,\phi\}} f_i(x)$$

où f_i, $1 \leq i \leq \phi$, est le i-ème facteur invariant de $xI - A$. Ce polynôme est multiple du polynôme minimal f_1 et divise le polynôme caractéristique χ_A. Comme au paragraphe 2.5 on note ν son degré. Au pire, le coût de son calcul est le même que celui du calcul de $\chi_{A'}$ précédemment. Par les techniques utilisées pour le lemme 2.5 dans [57] on peut certainement certifier que l'on a bien calculé un diviseur sur $\mathbb{Z}[x]$ du polynôme caractéristique. À partir du calcul de la forme normale de Frobenius de A modulo un nombre premier p, le polynôme χ'_A permet d'appliquer une remontée multi-facteur de Hensel [51, §15.5]. Nous poursuivons là l'idée de Storjohann pour le calcul de la forme de Frobenius d'une matrice creuse [115]. Cette remontée conduit à la forme normale de Frobenius de A. Le surcoût est principalement dû à la remontée elle-même. Les coefficients de n'importe quel diviseur de χ_A ont leurs tailles bornées par $l = \tilde{O}(n \log \|A\|)$ (voir par exemple [90] ou [57, Lemma 2.1]). Le coût de la remontée est donc en $\tilde{O}(\mathrm{M}(n)\mathrm{M}(l)) = \tilde{O}(\mathrm{M}(n)\mathrm{M}(n\log\|A\|))$ [51, Theorem 15.18].

Corollaire 4.3.2. *La forme normale de Frobenius et donc le polynôme minimal et le polynôme caractéristique de $A \in \mathbb{Z}^{n\times n}$ peuvent être calculés en $\tilde{O}\left(n^{2.7}\log\|A\|\right)$ opérations binaires par un algorithme probabiliste Monte Carlo. Le coût est $\tilde{O}\left(n^{3+1/5}\log\|A\|\right)$ sans produit rapide de matrices.*

Ceci représente un progrès sur l'algorithme de Giesbrecht et Storjohann [57]. Bien qu'aussi probabiliste, cette dernière méthode est cependant Las Vegas et peut être meilleure en arithmétique standard. Étant donné qu'il repose sur le calcul du polynôme minimal (voir les commentaires du §6.1), notre algorithme paraît résolument de type Monte Carlo.

Le résultat de Baur et Strassen basé sur la différentiation du déterminant et utilisé au paragraphe 3.3 pour le calcul de l'adjointe, ne peut être appliqué ici directement en complexité binaire. La borne (4.1) qui donne $\tilde{O}\left(n^{\omega+1}\log\|A\|\right)$ reste donc la seule connue pour le coût de l'inversion d'une matrice entière. La taille de la sortie n'est pourtant qu'en $\tilde{O}\left(n^3\log\|A\|\right)$. En se restreignant aux matrices polynomiales, nous abordons la question au prochain chapitre (§5.4).

5. Matrices sur $\mathsf{K}[x]$

Les deux expressions “polynômes matriciels” et “matrices polynomiales” pour parler des mêmes objets illustrent le double intérêt des matrices sur $\mathsf{K}[x]$. Les polynômes matriciels dans $\mathcal{M}_n[x]$ ont des zéros (les racines de leur déterminant). Leur étude est par exemple reliée aux problèmes de valeurs propres généralisées et à l'algorithmique des faisceaux de matrices. De nombreux outils disponibles pour les polynômes scalaires se généralisent au cas matriciel. On a vu que l'on dispose de la notion de pgcd et éventuellement de l'algorithme d'Euclide (§2.5). On a aussi la notion de résultant (voir notamment pour le calcul de formes normales [82, 123]) et la notion de fonction arithmétique (§2.2). Quant aux matrices polynomiales dans $\mathsf{K}[x]^{n\times n}$, comme matrices avant tout, leur étude algorithmique est reliée à celle des matrices en général.

Ici comme au chapitre précédent, ce n'est que récemment que l'on a pu passer en deçà de la borne indiquée par

$$\text{coût sur } \mathsf{K}[x] \leq \text{ coût arithmétique sur } \mathsf{K} \times \text{ degré maximum en sortie,} \tag{5.1}$$

c'est-à-dire en deçà de $n^\omega \times nd = n^{\omega+1}d$ (ou $n^4 d$) pour les problèmes de base tels que le déterminant pour une matrice de degré d en entrée (comparer à (4.1)). On peut déjà voir facilement que les techniques utilisées sur $\mathbb{Z}$ s'appliquent sur $\mathsf{K}[x]$. Il est plus intéressant de noter que les polynômes étant en général plus faciles à manier que les entiers (questions de retenues), les résultats pour l'instant connus vont même plus loin — par rapport à (4.1) et à (5.1) — sur $\mathsf{K}[x]$ que sur $\mathbb{Z}$. Les théorèmes 5.2.1, 5.3.1 et 5.4.1 n'ont par exemple pas encore leur équivalent sur $\mathbb{Z}$, même si pour certains problèmes, ce ne devrait être que temporaire (voir notamment la note de bas de page 27).

Les nouvelles complexités sur $\mathsf{K}[x]$ enlèvent un facteur n au terme de gauche de (5.1) pour aboutir à $\tilde{O}(n^\omega d)$ ou $\tilde{O}(n^3 d)$. C'est-à-dire que les problèmes se rapprochent du produit de matrices polynomiales de degrés d (cf remarque 5.4.5). C'est le cas du calcul du déterminant (théorème 5.2.1). Nous allons établir que c'est aussi le cas de la réduction en colonnes (base réduite d'un module sur $\mathsf{K}[x]$) et de l'inversion générique.

En travaillant sur les deux aspects complémentaires $\mathcal{M}_n[x]$ et $\mathsf{K}[x]^{n\times n}$ nous proposons deux nouvelles méthodes de calcul. Au paragraphe 5.3 nous donnons un algorithme pour la réduction en colonnes et pour le calcul de la forme normale de Popov. Cet algorithme s'exécute en $\tilde{O}(n^3 d)$ voire $\tilde{O}(n^\omega d)$ opérations sur K. Au paragraphe 5.4 nous exposons une approche pour l'inversion de matrices. Si la matrice en entrée est générique, c'est-à-dire qu'elle n'est pas sur une certaine sous-variété de dimension strictement inférieure à $n^2(d+1)$ de l'ensemble des matrices $n \times n$ de degré d, cette approche conduit à un calcul d'inverse en $\tilde{O}(n^3 d)$ opérations sur K. Ce coût étant à un facteur poly-logarithmique près la taille $n^2 \times nd$ de la sortie, l'algorithme est quasi-optimal. La méthode est en cours d'étude pour traiter le cas général et pour s'appliquer au cas entier. Son idée de base est de combiner l'algorithme récursif pour le pgcd de polynômes à une diagonalisation récursive de matrices. C'est une première avancée sur la borne (5.1) pour le problème de l'inverse.

Le calcul de l'inverse se fonde sur la notion de base minimale d'un module. On s'intéresse au paragraphe 5.1 à obtenir de telles bases quand le module est le noyau d'une matrice. Le paragraphe 5.2 reprend ensuite quelques points de calcul de formes normales de matrices polynomiales. On donne en particulier le résultat récent de Storjohann pour le calcul du déterminant en temps $\tilde{O}(n^\omega d)$ sur lequel se fonde notre réduction en colonnes.

5.1 Calculs de bases minimales de noyaux

On considère une matrice A de degré d dans $\mathsf{K}[x]^{m\times n}$ dont les lignes ne sont pas de plein rang *i.e.* $r = \text{rang}A < m$. On s'intéresse au noyau de A à gauche vu comme $\mathsf{K}[x]$-module, donc en particulier à l'ensemble des matrices $N(x) \in \mathsf{K}[x]^{(m-r)\times m}$ telles que

$$N(x)A(x) = 0 \in \mathsf{K}[x]^{(m-r)\times n}.$$

On a vu au théorème 1.4.1 et à l'exemple page 4 que la structure de ce module se caractérise par les degrés de ses bases minimales [45].

Définition 5.1.1. *Les indices de Kronecker (à gauche) de $A \in \mathsf{K}[x]^{m\times n}$ de rang r sont les degrés $\beta_1, \ldots, \beta_{m-r}$ des lignes d'une base réduite en colonnes de son noyau.*

Ces indices à gauche et à droite pour un faisceau de matrices $A = B + xC$ sont des invariants révélés par la forme normale de Kronecker. Dans ce cas ils se calculent en temps $O(m^2n)$ [9]. Une matrice $n \times n$ de degré d pouvant se linéariser en une matrice $nd \times nd$ — comme quand on associe une matrice compagnon à un polynôme — les indices de Kronecker se calculent en temps $O(n^3d^3)$ si $m = O(n)$ [10]. La linéarisation augmente quelque peu artificiellement l'exposant de d, on peut procéder autrement. Si U est une transformation unimodulaire de A vers une forme réduite en lignes (théorème 1.3.1), on a

$$UA = \begin{bmatrix} U_1 & U_2 \\ N_1 & N_2 \end{bmatrix} \begin{bmatrix} A_1 \\ A_2 \end{bmatrix} = \begin{bmatrix} A_r \\ 0 \end{bmatrix} \tag{5.2}$$

où le découpage des matrices U et A à gauche correspond au découpage en r et $m - r$ lignes à droite. Donc $N = [N_1\ N_2]$ est une base (non forcément réduite du noyau). L'identité (5.2) fait apparaître une autre interprétation de N. Le fait que

$$\begin{bmatrix} N_1 & N_2 \end{bmatrix} \begin{bmatrix} A_1 \\ A_2 \end{bmatrix} = \begin{bmatrix} 0 \end{bmatrix} \tag{5.3}$$

montre que N_1 et N_2 jouent le rôle de cofacteurs tels que $N_1A_1 + N_2A_2 = 0$. Dans le cas de polynômes scalaires de degrés d, on sait que l'on peut trouver des cofacteurs de degrés au plus d. Cela peut être faux dans le cas scalaire et seule la somme des degrés des lignes des cofacteurs est bornée *a priori* par (voir §1.4)

$$\sum_{i=1}^{m-r} \beta_i \leq rd.$$

D'après (5.2), une première alternative à l'approche faisceau pour calculer une base minimale du noyau est de calculer une base "quelconque" à l'aide d'une transformation matricielle U (voir par exemple [114, §6.2]). Cela consiste ensuite à réduire cette base par une méthode de réduction en lignes (voir [124, 94] et les références qui y sont données, voir aussi §5.3).

Avec M.P. Quéré-Stuchlik nous avons proposé de calculer directement une base réduite par un algorithme adaptatif en les indices de Kronecker. Il se base sur des algorithmes existants d'approximation matricielle. L'idée est la même que celle de la reconstruction d'une fraction rationnelle exacte par un algorithme d'approximation par exemple de type Padé-Hermite. Par identification des termes le lemme suivant est trivial.

Lemme 5.1.2. *Si $N(x)$ est de degré inférieur à β et vérifie l'approximation $N(x)A(x) = O(x^{\beta+d+1})$ alors $N(x)A(x) = 0$.*

Une base minimale peut alors être calculée comme sous-matrice d'une base d'approximants $P(x)A(x) = O(x^\sigma)$ si l'ordre d'approximation σ est assez élevé. En utilisant l'algorithme de Beckermann & Labahn qui calcule une base dite σ-base [5], on arrive à la complexité suivante.

Proposition 5.1.3. *Si les indices de Kronecker sont bornés par β alors une base minimale du noyau de $A \in \mathsf{K}[x]^{m\times n}$ avec $m = O(n)$ se calcule en $O(n^2\mathrm{M}(n\beta)\log(n\beta))$ (voir [105, Chap. 4], [7, §4.2], [5] et comparer au théorème 1.5.3).*

L'aspect adaptatif provient du fait que si la base minimale d'approximants est calculée à l'ordre $\beta + d + 1$, alors d'après le lemme 5.1.2, elle contient forcément une base du noyau. Puisque β est en $O(nd)$ au pire, cela donne jusqu'à $\tilde{O}(n^4 d)$ au total. La taille de la base n'est pourtant qu'en $O(n\sum \beta_i) = O(n^2 d)$. Ce coût, classique dans le cas général (c'est (5.1)), sera intéressant pour le calcul de bases plus particulières comme pour l'inversion générique au paragraphe 5.4.

5.2 Formes normales

Les formes normales de matrices polynomiales telles que celles de Popov, d'Hermite ou de Smith ainsi que des matrices de passage associées se calculent toutes en temps $\tilde{O}(n^{\omega+1}d)$. On pourra se référer à [124, 114] et aux diverses références qui y sont données. Avec B. Beckermann et G. Labahn, nous avons aussi développé la notion de forme normale à décalage et généralisé l'algorithme adaptatif du paragraphe précédent à leur calcul [8, 7].

Pour le calcul du déterminant et de la forme normale de Smith (sans transformation associée), l'exposant de n peut être diminué de un.

Théorème 5.2.1. *La forme normale de Smith et donc le déterminant d'une matrice $A \in \mathsf{K}[x]^{n\times n}$ de degré d se calculent par un algorithme probabiliste Las Vegas en $\tilde{O}(n^\omega d)$ opérations dans K [116, Proposition 24].*

On n'est pas encore à l'optimalité puisqu'*a priori* on sait seulement minorer le coût par $O(n^\omega + n^2 d)$, ce qui est le coût du produit de matrices plus la taille de l'entrée. Le coût du théorème est néanmoins très faible puisque équivalent au meilleur coût connu pour la résolution d'un système linéaire (théorème 1.9.1) et inférieur à la taille de A^{-1} au pire. La méthode conduisant à ce coût est due à Storjohann. Nous en avons interprété une des clefs avec la proposition 1.9.3 (cf aussi §5.3 ci-dessous) et en avons dégagé le lien avec l'approche Krylov / Lanczos par blocs en page 17.

On l'a dit en introduction, les coûts du chapitre précédent sur $\mathbb{Z}$ se transfèrent naturellement à $\mathsf{K}[x]$. On a par exemple cet autre résultat à partir du corollaire 4.3.2.

Corollaire 5.2.2. *Le polynôme caractéristique de $A \in \mathsf{K}[x]^{n\times n}$ se calcule en $\tilde{O}(n^{2.7}d)$ opérations dans K par un algorithme probabiliste Monte Carlo. Le coût est $\tilde{O}(n^{3+1/5}d)$ sans produit rapide de matrices.*

Comme dans tout ce chapitre, une première cible à atteindre pour le polynôme caractéristique reste donc en $n^\omega d$. Sans produit rapide de matrices ou de polynômes ni calcul des matrices de passage, cette cible est atteinte pour les formes normales d'Hermite et de Popov. Le théorème qui suit donne un gain sur l'exposant de n et un gain global en arithmétique standard.

Théorème 5.2.3. *La forme normale d'Hermite et la forme normale de Popov (dans le cas plein rang) d'une matrice $A \in \mathsf{K}[x]^{n\times n}$ se calculent en $O(n^3 d^2)$ opérations sur K [94].*

5.3 Réduction en colonnes et forme normale de Popov

L'exposant du théorème 5.2.3 peut encore être amélioré pour le calcul d'une forme réduite en colonnes et de la forme normale de Popov (les définitions ont été vues au §1.3). Pour simplifier on ne traite ici que les matrices $A \in \mathsf{K}[x]^{n\times n}$ non singulières de degré d.

La réduction en colonnes est essentiellement une réduction de base de réseau dans le cas polynomial. On appelle M le $\mathsf{K}[x]$-module engendré par les colonnes de A. Il s'agit de calculer une matrice $C \in \mathsf{K}[x]^{n\times n}$ dont les colonnes forment une base minimale de M (définition 1.4.1). On doit avoir en particulier que

$$C = AU \in \mathsf{K}[x]^{n\times n}$$

avec U unimodulaire. On se propose de combiner notre technique de [124] aux résultats de [116]. L'idée est de se ramener à un calcul d'approximation de Padé matricielle qui a pour "effet de bord" de normaliser les matrices. À A^{-1} on associe la description irréductible (R, A) d'une fraction

$$H = RA^{-1} \tag{5.4}$$

strictement propre, c'est-à-dire tendant vers 0 quand x tend vers l'infini. En particulier, les degrés des colonnes du numérateur R sont strictement inférieurs aux degrés des colonnes correspondantes du dénominateur A. Tous les degrés sont donc plus petits que d. Si H se développe en l'infini sous la forme

$$H(x) = \sum_{i\geq 0} \frac{H_i}{x^{i+1}} \in \mathsf{K}(x)^{n\times n}$$

alors un approximant de Padé matriciel $(\bar{R}, C)$ — calculé à partir des $2d$ premiers termes de la série — vérifie

$$H = \bar{R}C^{-1} \tag{5.5}$$

avec C sous forme réduite en colonnes. Par unicité de la fraction irréductible à normalisation près (théorème 1.6.5), il suit que $C = AU$ avec U unimodulaire et que C est une forme réduite en colonnes de A. Cela conduit à l'algorithme et à la justification qui suivent.

Algorithme Réduction en colonnes
Entrée : $A \in \mathsf{K}[x]^{n\times n}$ de degré d.
Sortie : $C = AU$ forme réduite en colonnes de A.

Choix de x_0 aléatoire dans K
Si $\det A(x_0) = 0$ alors échec /* *Retirage de* x_0 *ou* A *est probablement singulière* */

$h := (n-1)d + 1$
$\sum_{i=0}^{2d-1} H_i/x^{i+1} :=$ les $2d$ premiers termes de $(A^{-1}(x+x_0) - (A^{-1}(x+x_0) \bmod x^h))/x^h$
Calcul d'un approximant tel que $\sum_{i=0}^{2d-1} H_i/x^{i+1} \equiv \bar{R}(x)C^{-1}(x) \bmod x^{2d}$
$C(x) := C(x - x_0)$.

En général, si A n'est pas réduite en colonnes, A^{-1} n'est justement pas strictement propre et l'on ne peut considérer directement la fraction A^{-1} c'est-à-dire prendre $R = I$ dans (5.4). On peut cependant appliquer l'algorithme x-adique de la proposition 1.9.3 avec $B = I$, $P = I$ et $Q = A$ pour calculer une matrice R adéquate. On a donc $m = n$ et Q vérifie bien la condition $m(\deg Q)/d = m = O(n)$. En outre, puisque A^{-1} est une fraction irréductible (de numérateur

l'identité), la proposition 1.9.3 assure que RA^{-1} est aussi irréductible. La description (5.4) est donc obtenue en $\tilde{O}(n^\omega d)$. En appliquant l'analogue de (1.15) en division à gauche, on sait que la même fraction admet une description irréductible à gauche $A^{-1}R'$.

À partir des $2d$ premiers termes du développement de RA^{-1} on peut alors calculer un dénominateur C éventuellement normalisé pour (5.5) par approximation. On se reportera au coût donné au théorème 1.5.3 et aux références citées juste après en page 6 puisque cela revient à calculer le polynôme minimal d'une suite.

Si $A(0) = 0$, on peut trouver un choix aléatoire de x_0 tel que $A(x_0) \neq 0$ et faire le calcul pour $A(x + x_0)$ puisqu'alors il suffit de prendre en résultat $C(x) = C(x - x_0)$. Cela donne donc un algorithme de type Las Vegas puisque tout est déterministe dès qu'un point régulier est trouvé pour l'algorithme x-adique.

Théorème 5.3.1. *La forme normale de Popov d'une matrice $A \in \mathsf{K}[x]^{n\times n}$ inversible se calcule en $\tilde{O}\left(n^3 d\right)$ opérations dans K par un algorithme probabiliste Las Vegas.*

Nous développons par ailleurs un algorithme de coût $\tilde{O}(n^\omega d)$ en apportant quelques modifications aux algorithmes d'approximation [132].

Exemple. Soit à calculer une forme réduite en colonnes de

$$A(x) = \begin{bmatrix} x+3 & x^3+3\,x^2+x+4 \\ -2 & -2\,x^2-1+x \end{bmatrix}.$$

Sa matrice inverse n'est pas strictement propre :

$$A^{-1}(x) = \begin{bmatrix} -\frac{2\,x^2+1-x}{x^2+4\,x+5} & -\frac{x^3+3\,x^2+x+4}{x^2+4\,x+5} \\ \frac{2}{x^2+4\,x+5} & \frac{x+3}{x^2+4\,x+5} \end{bmatrix}$$

mais d'après (1.15) on peut la décomposer en une partie polynomiale et une partie fractionnaire

$$\begin{aligned} A^{-1}(x) = &\ (A^{-1}(x) \bmod x^4) + x^4 R(x) A^{-1}(x) \\ = &\ (A^{-1}(x) \bmod x^4) + x^4 \begin{bmatrix} -\frac{279}{625} & -\frac{99}{125} - \frac{279}{625}\,x^2 \\ \frac{48}{625} & \frac{13}{125} + \frac{48}{625}\,x^2 \end{bmatrix} \begin{bmatrix} x+3 & x^3+3\,x^2+x+4 \\ -2 & -2\,x^2-1+x \end{bmatrix}^{-1}. \end{aligned}$$

On a maintenant une fraction RA^{-1} strictement propre dont le développement permet de calculer un approximant normalisé

$$R(x)A^{-1}(x) \ = \ \bar{R}(x)C^{-1}(x) = \begin{bmatrix} -\frac{279}{625} & -\frac{216}{625} \\ \frac{48}{625} & \frac{17}{625} \end{bmatrix} \begin{bmatrix} x+3 & 1 \\ -2 & x+1 \end{bmatrix}^{-1}.$$

La matrice dénominateur C de cet approximant est la forme normale de Popov de A. □

5.4 Inversion

À l'image de la situation sur $\mathbb{Z}$ (voir les commentaires en fin du chapitre 4) tous les progrès autour du déterminant, de la forme normale de Smith et même du polynôme caractéristique, ne semblent rien donner pour l'inversion. À nouveau sur $\mathsf{K}[x]$, on peut dire que l'algorithme algébrique de Baur & Strassen, qui calcule l'inverse par différentiation du déterminant, fait défaut. Avec C.P. Jeannerod, nous proposons une nouvelle méthode d'inversion qui atteint le coût quasi-optimal de $\tilde{O}\left(n^3 d\right)$ pour une matrice générique en entrée [133, 63]. Plus précisément, l'algorithme a cette

complexité sauf pour des matrices sur une sous-variété de dimension strictement inférieure à $n^2(d+1)$. On cherche à imbriquer complètement opérations matricielles et opérations polynomiales pour éviter un coût global donné par le produit d'un coût matriciel et d'un coût polynomial.

Exemple. Soit A la matrice carrée de dimension 4 et de degré $d = 2$ sur $\mathsf{K} = \mathbb{Z}/3\mathbb{Z}$:

$$A = \begin{bmatrix} x+2 & x^2+2x+2 & 2x^2+2x+2 & 2x^2+2x+2 \\ x+2 & 1 & 2x+1 & 2x^2+2x+2 \\ x^2+2x+2 & 2+x^2 & 2x^2 & 0 \\ 1 & x & x^2+x+1 & 2x^2 \end{bmatrix}.$$

La première étape de l'algorithme consiste à appliquer (5.3) deux fois. On calcule des cofacteurs minimaux $[N_1\ N_2]$ pour l'ensemble des $n/2$ premières colonnes de A et des cofacteurs minimaux $[M_1\ M_2]$ pour les $n/2$ dernières colonnes. Ces cofacteurs permettent de former par exemple la matrice

$$U^{(1)}(x) = \begin{bmatrix} M_1(x) & M_2(x) \\ N_1(x) & N_2(x) \end{bmatrix} = \begin{bmatrix} 2x^2 & x^2 & 2+x^2 & 0 \\ 2+x^2+x & 1+2x+x^2 & 2x+1 & x^2+x+1 \\ x+2 & 1+x^2 & 2+2x & 2+2x \\ 2x & x+2 & 2 & 1+x^2 \end{bmatrix}$$

telle que

$$U^{(1)}(x)A(x) = \begin{bmatrix} 2x^3+x^2+x+1+x^4 & x^3+1 & 0 & 0 \\ x+x^3+x^2 & x+1+x^4 & 0 & 0 \\ 0 & 0 & x^3+1 & 2x^3+2x^2+2x+2x^4 \\ 0 & 0 & 2x^3+x+x^4 & x+1+2x^4 \end{bmatrix}.$$

On voit que cela permet de diagonaliser par blocs $(n/2)\times(n/2)$ à l'aide d'une matrice de transformation de degré seulement $d = 2$. Deux étapes d'élimination de Gauss auraient au contraire fait intervenir un déterminant 2×2 de degré 4. L'algorithme d'inversion procède ensuite itérativement pour arriver à une matrice diagonale et finalement à l'identité. □

L'algorithme complet est donné un peu plus loin en supposant que la dimension de la matrice est une puissance de deux. Il est déterministe et retourne l'inverse pour toute matrice inversible en entrée.

Nous verrons ensuite certains des résultats utilisés pour sa preuve et pour l'étude de son coût. Ce coût dépend *a priori* de la matrice en entrée. Ce qui va nous intéresser, c'est de montrer que pour presque toutes les matrices, l'algorithme s'exécute en $\tilde{O}(n^3 d)$ opérations dans K. Pour des matrices particulières (conduisant à des cofacteurs de degrés élevés) il semble que le coût puisse être bien supérieur à ces quantités c'est-à-dire sans doute au mieux majoré comme $\tilde{O}(n^{\omega+1}d)$.

Théorème 5.4.1. *Pour toute matrice $A \in \mathsf{K}[x]^{n\times n}$ de degré d sauf sur une sous-variété de dimension strictement inférieure à $n^2(d+1)$, l'algorithme* Inverse *calcule l'inverse de A en $\tilde{O}(n^3 d)$ opérations dans K.*

La matrice A en entrée est diagonalisée en $\log n$ étapes. À l'étape i l'algorithme construit par récurrence une matrice $U^{(i)} \in \mathsf{K}[x]^{n\times n}$ telle que $U^{(i)}U^{(i-1)}\dots U^{(1)}A$ soit diagonale par blocs, avec 2^i blocs $2^{l-i}\times 2^{l-i}$. Les matrices $U^{(i)}$ sont accumulées dans une matrice U et la matrice diagonale

par blocs qui en résulte est chaque fois stockée dans une même matrice B. L'inverse de A est donc finalement obtenue comme produit de U et de l'inverse de la matrice diagonale stockée dans B à la dernière étape. Pour toute matrice M carrée $m \times m$ on note M_L et M_R les sous-matrices formées de ses $m/2$ premières et $m/2$ dernières colonnes. On suppose aussi disposer d'une fonction $\mathsf{BaseMin}$ qui calcule une base minimale du noyau à gauche d'une matrice polynomiale quelconque (cf §5.1).

Algorithme $\mathsf{Inverse}$
Entrée : $A \in \mathsf{K}[x]^{n\times n}$ de degré d, $n = 2^l$.
Sortie : A^{-1} si A est inversible.

$B := A$
$U := I$

Pour i de 1 à l faire /* B *est diagonale par blocs de blocs* $B^{(k)}$, $1 \le k \le 2^{i-1}$ */
  pour k de 1 à 2^{i-1} faire
    $\overline{U}^{(k)} := \mathsf{BaseMin}(B_R^{(k)})$
    $\underline{U}^{(k)} := \mathsf{BaseMin}(B_L^{(k)})$

  $$U^{(i)} := \operatorname{diag}\left(\begin{bmatrix} \overline{U}^{(1)} \\ \underline{U}^{(1)} \end{bmatrix}, \ldots, \begin{bmatrix} \overline{U}^{(2^{i-1})} \\ \underline{U}^{(2^{i-1})} \end{bmatrix} \right)$$

  $B := U^{(i)}B$
  $U := U^{(i)}U$
$A^{-1} := B^{-1}U$.

On note D la matrice diagonale B à la fin de l'étape l. La dernière affectation de l'algorithme donne

$$D = UA \in \mathsf{K}[x]^{n\times n}$$

avec $U \in \mathsf{K}[x]^{n\times n}$, c'est-à-dire que D est une matrice diagonale multiple de A. L'algorithme calcule la matrice inverse sous la forme factorisée

$$A^{-1} = D^{-1}U^{(\log n)}U^{((\log n)-1)} \ldots U^{(1)}. \tag{5.6}$$

Tout en rappelant le type de factorisation que l'on peut obtenir par applications récursives du complément de Schur, cette décomposition a en fait une complexité bien plus faible. Elle revient à manipuler des fractions matricielles irréductibles et donc de degrés inférieurs à ceux des descriptions habituellement utilisées dans une élimination de Gauss par blocs. La propriété fondamentale pour borner le coût de l'algorithme est donnée par le lemme suivant.

Proposition 5.4.2. *Pour toute matrice A sauf sur une sous-variété, les matrices $U^{(i)}$ sont diagonales par blocs, avec 2^{i-1} blocs $2^{l-i+1} \times 2^{l-i+1}$, et sont de degrés exactement $2^{i-1}d$ [63].*

Cette proposition peut se démontrer par récurrence à partir du résultat suivant :

Lemme 5.4.3. *Pour $M(x) = M_0 + xM_1 + \ldots + x^dM_d \in \mathsf{K}[x]^{2m\times m}$ de degré d, la matrice Toeplitz par blocs carrée définie par*

$$\mathcal{T}(M) = \begin{bmatrix} M_0 & M_1 & \ldots & M_d & & \\ & \ddots & \ddots & & \ddots & \\ & & M_0 & M_1 & \ldots & M_d \end{bmatrix} \in \mathsf{K}^{2md\times 2md} \tag{5.7}$$

est inversible si et seulement si les indices de Kronecker de M à gauche (les degrés d'une base minimale de son noyau) sont $\beta_1 = \beta_2 = \ldots = \beta_m = d$.

Ceci provient de l'identité (1.7) sur le degré de McMillan $\delta(M)$ de M. Le fait que la matrice $\mathcal{T}(M)$ est inversible est équivalent au fait que tout indice de Kronecker est supérieur à d et donc au fait que

$$md + \sum_{j=1}^{r} d_j \leq (2m-r)d + \sum_{j=1}^{r} d_j \leq \sum_{l=1}^{m-r} \beta_l + \sum_{j=1}^{r} d_j = \delta(A) \leq md,$$

ce qui conduit à $r = m$ et au résultat voulu puisque tout est positif. La récurrence pour prouver la proposition 5.4.2 utilise le fait suivant.

Lemme 5.4.4. *Soient $n^2 d$ indéterminées $\alpha_{i,j,k}$ pour $1 \leq i, j \leq n$ et pour $1 \leq k \leq d$. Il existe un polynôme non nul $\Delta \in \mathsf{K}(\alpha_{1,1,0}, \alpha_{1,1,1}, \ldots, \alpha_{i,j,k}, \ldots, \alpha_{n,n,d})$ tel que si Δ est non nul lorsque évalué en les coefficients $a_{i,j}(x) = a_{i,j,0} + a_{i,j,1}x + \ldots + a_{i,j,d}x^d$ d'une matrice $A \in \mathsf{K}[x]^{n\times n}$ de degré d, alors l'assertion de la proposition 5.4.2 est vraie.*

Ce second lemme s'obtient en appliquant d'abord le premier lemme 5.4.3 à chacune des $\log n$ étapes de l'algorithme. À l'étape i, 2^i calculs de bases minimales sont en jeu pour les matrices $B_L^{(k)}$ et $B_R^{(k)}$, $1 \leq k \leq 2^{i-1}$. La construction de bases normalisées uniques (sous forme de Popov) permet de bien définir des déterminants $\Delta_L^{(i,k)}$ et $\Delta_R^{(i,k)}$ de matrices Toeplitz $\mathcal{T}(B_L^{(k)})$ et $\mathcal{T}(B_R^{(k)})$. Si, pour une matrice donnée, ces bases normalisées à l'étape i ont les bons degrés $2^{i-1}d$ alors n'importe quelle base minimale aura les bons degrés. Il suffit alors de prendre

$$\Delta = \prod_{i=1}^{\log n} \prod_{k=1}^{2^{i-1}} \Delta_L^{(i,k)} \Delta_R^{(i,k)}.$$

On démontre ensuite que Δ n'est pas identiquement nul en explicitant une matrice polynomiale qui ne peut en être racine. La sous-variété du théorème 5.4.1 et de la proposition 5.4.2 est donc celle définie par Δ.

Une fois démontrée la propriété sur les degrés des bases minimales successives construites par l'algorithme, le coût annoncé au théorème 5.4.1 s'obtient directement. Il est le cumul d'une part du coût du calcul des bases minimales et d'autre part de la mise à jour de la matrice courante et de l'inverse. Ce à quoi il faut ajouter l'inversion finale d'une matrice diagonale et le produit subséquent. D'après la proposition 5.1.3 sur le calcul des bases minimales, à l'étape i, les 2^i bases — aux degrés $2^{i-1}d$ connus *a priori* — sont obtenues en temps

$$\tilde{O}\left(2^i \times \left(2^{2(l-i)}\mathrm{M}(2^{l-i}2^{i-1}d)\right)\right) = \tilde{O}\left(2^{-i}n^2\mathrm{M}(nd)\right). \tag{5.8}$$

La multiplication $U^{(i)}B$ est la multiplication de deux matrices de même structure. Elles ont 2^{i-1} blocs $2^{l-i+1} \times 2^{l-i+1}$ de degrés $2^{i-1}d$. Leur produit est donc de coût

$$\tilde{O}\left(2^{i-1} \times (2^{l-i+1})^\omega \times \mathrm{M}(2^{i-1}d)\right) = \tilde{O}\left((2^i)^{2-\omega}n^\omega d\right).$$

Dans le deuxième produit à l'étape i il faut multiplier $U^{(i)}$ et U qui est une matrice dense $n \times n$ de degré 2^{i-2} (de degré 0 pour $i = 1$). En effectuant ce produit par blocs on arrive à

$$\tilde{O}\left(2^{i-1} \times 2^{i-1} \times (2^{l-i+1})^\omega \times \mathrm{M}(2^{i-1}d)\right) = \tilde{O}\left((2^i)^{3-\omega}n^\omega d\right). \tag{5.9}$$

Ce dernier coût correspond aussi à celui qui met en jeu la matrice diagonale finale puisque c'est une matrice diagonale de degré nd.

En récapitulant, on voit qu'au début de l'algorithme, l'étape de calcul des bases minimales domine, l'identité (5.8) donne $\tilde{O}\left(n^2\mathrm{M}(nd)\right)$. En fin d'algorithme, c'est la mise à jour de l'inverse qui coûte le plus puisque l'on a aussi $\tilde{O}\left(n^2\mathrm{M}(nd)\right)$ dans (5.9).

La sous-variété définie par Δ n'est pas la meilleure possible en ce sens que l'algorithme s'exécutera en $\tilde{O}\left(n^3d\right)$ dans bien d'autre cas. Sa construction, qui passe par des bases normalisées, le laisse voir. Nous poursuivons les études dans cette direction ainsi que pour fournir un calcul d'inverse dans le cas général.

En conclusion de ce chapitre, les différents coûts présentés amènent un commentaire plus général.

Remarque 5.4.5. Une conséquence immédiate au théorème 1.1.1 est que les problèmes que nous avons regardés sur $\mathsf{K}[x]$ ne peuvent être résolus qu'en temps $\Omega(n^\omega)$. Les complexités obtenues sur $\mathsf{K}[x]$ sont $\tilde{O}\left(n^\omega d\right)$ pour le déterminant et la forme normale de Smith et $\tilde{O}\left(n^3d\right)$ pour la réduction en colonnes et l'inversion. Cela suggère la question du positionnement de ces problèmes par rapport à celui du calcul du produit de deux polynômes matriciels de degrés d.

6. Matrices boîtes noires

Le coût principal de l'approche Krylov / Lanczos est celui de multiplications matrice × vecteur ou matrice × matrice pour produire un sous-espace de Krylov $\langle Y, AY, A^2Y \ldots \rangle$ ou une suite matricielle $XY, XAY, XA^2Y, \ldots$ (voir le chapitre 2). Les matrices A les plus concernées sont donc notamment celles pour lesquelles le produit par un vecteur est peu onéreux comparé à $O(n^2)$. C'est ce qui conduit à utiliser la notion de boîte noire — apparue au départ pour des polynômes [75] — c'est-à-dire à considérer la matrice seulement comme un endomorphisme abstrait. Le coût de l'application de l'endomorphisme à un vecteur sur un corps K, noté $\mathrm{E}(n)$ en dimension n, sert d'unité. Les matrices structurées et les matrices creuses par exemple, entrent dans la catégorie des matrices telles que $\mathrm{E}(n) = o(n^2)$. La notion de boîte noire permet d'abstraire les structures particulières. L'intérêt porté au sujet vient du fait que la plupart des matrices issues de domaines applicatifs se prête bien à cette modélisation.

Du point de vue du calcul exact, on se pose la question de la complexité des problèmes en fonction de $\mathrm{E}(n)$ (voir [68, §3]). L'objectif principal est une résolution en $O(n)$ ou $\tilde{O}(n)$ produits matrice × vecteur, puisqu'alors $\mathrm{E}(n) = o(n^2)$ donne un coût total en $\tilde{o}(n\mathrm{E}(n)) = \tilde{o}(n^3)$ soit moins d'opérations que pour une résolution générale en, mettons, $\tilde{O}(n^3)$. Pour des boîtes noires plus particulières avec $\mathrm{E}(n) = \tilde{O}(n)$ (e.g. des matrices très creuses), un objectif intermédiaire est de parvenir à une résolution en $o(n^2)$ produits matrice × vecteur. Ce dernier cas conduisant en effet aussi à un coût total en $o(n^2\mathrm{E}(n)) = \tilde{o}(n^3)$.

Nous sommes intervenu sur les deux objectifs pour d'une part systématiser l'utilisation du polynôme minimal matriciel et améliorer les réductions qu'il permet et d'autre part progresser sur la question ouverte du calcul du polynôme caractéristique (problème 6.4.1).

Le paragraphe 6.1 reprend les complexités de base avec la question centrale du calcul du polynôme minimal. Comme on a déjà pu le voir avec le déterminant, plusieurs problèmes se réduisent à ce calcul pour autant que la matrice en entrée remplisse certaines conditions. Si ces conditions ne sont pas remplies *a priori*, il est possible de pré-multiplier la matrice — de la pré-conditionner — pour que les conditions soient satisfaites par la nouvelle matrice. En termes de boîtes noires le produit n'est jamais effectué mais on manipule une boîte noire composée (voir (6.3)). Cette notion de pré-conditionnement et les manières dont on l'applique au déterminant, au rang et à la résolution de systèmes linéaires, sont le sujet des paragraphes 6.2 et 6.3. Calculer le polynôme caractéristique — ou calculer toutes les valeurs propres avec leurs multiplicités — est un problème qui en l'état actuel des choses est plus difficile que de seulement calculer le polynôme minimal. Nous exposons au paragraphe 6.4 la méthode qui nous a permis un progrès sur le sujet et concluons au paragraphe 6.5 par quelques commentaires plus généraux.

Mis à part le dernier paragraphe, nous exprimons les résultats pour un corps K abstrait. La plupart des questions a en réalité été abordée pour des corps finis, les travaux sur $\mathbb{Q}$ ont été moins nombreux. Beaucoup de détails techniques non donnés ici, notamment dans le cas des corps à peu d'éléments, concerneraient les études des probabilités de succès des algorithmes (cf §2.2).

6.1 Approche par boîtes noires

Pour deux vecteurs aléatoires $x \in \mathsf{K}^{1\times n}$ et $y \in \mathsf{K}^{n\times 1}$ le polynôme minimal de la suite xA^iy, $i \geq 0$, se calcule à partir de ses $2n$ premiers termes (théorème 1.5.3). Avec une bonne probabilité

(fonction Φ du paragraphe 2.2), ce polynôme coïncide avec le polynôme minimal π_A de A.

Théorème 6.1.1. *Le polynôme minimal de $A \in \mathsf{K}^{n\times n}$ se calcule par un algorithme probabiliste Monte Carlo en $O(n\mathrm{E}(n) + n^2)$ opérations dans K [134, 74].*

Par terminaison anticipée (voir la discussion en fin du §2.5), l'algorithme peut être rendu adaptatif en le degré d de π_A pour un coût de $O(d\mathrm{E}(n) + dn)$ opérations, éventuellement dans une extension de K pour disposer de suffisamment d'éléments (voir §2.5). Quant aux constantes du "O" au théorème 6.1.1, pour ϵ fixé aussi petit que l'on veut, on peut atteindre $((1+\epsilon)n + O(1))\mathrm{E}(n) + \tilde{O}(n^2)$ avec une technique par blocs [67, §7]. On ne sait pas encore certifier le polynôme minimal avec un coût aussi faible que $O(n\mathrm{E}(n) + n^2)$ et donc concevoir un algorithme Las Vegas. Cela demanderait de faire intervenir tout l'espace et non pas seulement quelques vecteurs pour les projections.

Dans le cas par blocs, pour X et Y aléatoires, le polynôme minimal matriciel $F_X^{A,Y} \in \mathsf{K}[x]^{m\times m}$ est de degré $\nu/m \leq n/m$ et se calcule en temps $\tilde{O}(m^\omega n/m) = \tilde{O}(m^{\omega-1}n)$ (théorèmes 2.1.1 et 2.5.1). Sa forme normale de Smith se calcule en $\tilde{O}(m^\omega n/m)$ (théorème 5.2.1). Le calcul du polynôme minimal scalaire $\pi_A(x) \in \mathsf{K}[x]$ se déduit donc sans surcoût du polynôme minimal matriciel $F_X^{A,Y} \in \mathsf{K}[x]^{m\times m}$.

Pour A inversible, en prenant x aléatoire et $y = b$ fixé, le polynôme minimal de xA^ib, $i \geq 0$, conduit avec une bonne probabilité à

$$f_0 b + f_1 Ab + \ldots + f_{d-1}A^{d-1}b + A^d b = 0,$$

soit à

$$A((-1/f_0)(f_1 b + \ldots + f_{d-1}A^{d-2}b + A^{d-1}b)) = b \tag{6.1}$$

et donc à la résolution du système linéaire $Az = b$. Quant aux aspects adaptatifs, les remarques restent les mêmes : on peut considérer que la projection à droite est aussi aléatoire en résolvant le système $Az = b + Au$ pour u un vecteur aléatoire [41, §4]. À la différence du polynôme minimal, il est ici facile de vérifier *a posteriori* que l'on a bien une solution.

Théorème 6.1.2. *La solution à $Az = b$ pour A inversible dans $\mathsf{K}^{n\times n}$ se calcule par un algorithme probabiliste Las Vegas en $O(n\mathrm{E}(n) + n^2)$ opérations dans K [134, 74].*

À partir des études de Lambert [83] et de Dumas [33] on peut préciser la comparaison entamée au §2.6 de l'approche de Wiedemann et de celle de Lanczos pour le calcul de π_A et pour la résolution d'un système linéaire. Les coûts dominants sont repris dans la table ci-dessous. Les quantités entre crochets donnent l'espace de stockage nécessaire (en nombres d'éléments de K). On suppose que l'on peut aussi appliquer la transposée de A à un vecteur en temps $\mathrm{E}(n)$.

	Terminaison anticipée π_A	Monte Carlo π_A	Las Vegas $Az = b$
Wiedemann [$6n$]	$2d\mathrm{E}(n) + 4d(n+d)$	$2n\mathrm{E}(n) + 4n^2 + 2d(n+d)$	$+d\mathrm{E}(n) + 2dn$
Wiedemann [$O(dn)$]	$2d\mathrm{E}(n) + 4d(n+d)$	$2n\mathrm{E}(n) + 4n^2 + 2d(n+d)$	$+2dn$
Lanczos [$3n$]	$2d\mathrm{E}(n) + 8dn$	$2n\mathrm{E}(n) + 4n^2 + 4dn$	$+2dn$

Table 1 [34, p. 46]. **Coûts comparés des approches de Wiedemann et de Lanczos**

Côté Lanczos, la terminaison anticipée et l'algorithme Monte Carlo correspondent à la version Lanczos bi-orthogonale avec ou sans *look-ahead* [83]. Pour les deux approches le nombre de produits

matrice × vecteur est divisé par deux si la matrice est symétrique. Comme la mise à jour du polynôme générateur de la suite est faite par produits scalaires au lieu d'opérations polynomiales élémentaires, la statégie Lanczos est un peu plus coûteuse pour le calcul de π_A. Dans le cas de la résolution de $Az = b$, séparer le calcul de π_A du calcul de z à partir de (6.1) est un désavantage. Cela demande soit de recalculer les itérés $(A^i b)_{0 \le i \le d-1}$, soit de les stocker. On aura le même type de conclusions pour les versions par blocs de ces algorithmes [33].

6.2 Pré-conditionnement algébrique

Le calcul du déterminant pages 17 et 29 reposait sur le fait que le polynôme caractéristique de A était égal à son polynôme minimal. Cette propriété n'est pas vraie en général mais on va voir qu'il est possible de l'assurer à moindre coût en pré-multipliant la matrice A par des matrices bien choisies (voir le théorème 6.2.1). Cette opération est appelée pré-conditionnement (algébrique) de A. On l'utilise pour modifier des propriétés algébriques de la matrice sans rapport *a priori* avec la notion numérique de nombre de conditionnement. Le procédé a été introduit par Wiedemann [134] puis systématiquement repris ensuite notamment pour calculer le rang ou pour la résolution de systèmes linéaires avec A non forcément inversible.

Dans [21, §3] nous avons identifié trois grandes classes de préconditionnements. La première concerne ceux qui assurent des propriétés d'indépendance linéaire. Par example, si A est de rang r, il faut assurer que les r premières colonnes de la nouvelle matrice A' sont indépendantes ou que ses r premiers mineurs principaux sont non nuls. La deuxième classe assure que la forme de Jordan de A' n'a pas de bloc nilpotent non trivial, c'est-à-dire que son polynôme minimal est de valuation 1. La troisième classe s'intéresse à des propriétés de cyclicité pour par exemple aboutir à ce que A' ait un polynôme minimal sans carré (en particulier, le sous-espace stable associé aux valeurs propres non nulles est cyclique).

Pour réaliser ces pré-conditionnements la technique générale consiste donc à pré-multiplier A à gauche et/ou à droite par deux matrices C_g et C_d :

$$A' = C_g A C_d. \tag{6.2}$$

Le choix de C_g et C_d est bien sûr guidé par la propriété cible à réaliser, mais aussi par le fait que le surcoût doit être le plus faible possible. On remarquera que dans l'approche Krylov / Lanczos le produit (6.2) n'a pas à être formé explicitement puisque pour calculer $(C_g A C_d)^i y$ on peut utiliser que

$$(C_g A C_d)^i y = C_g(A(C_d((C_g A C_d)^{i-1} y))). \tag{6.3}$$

Les matrices C_g et C_d peuvent donc aussi être manipulées comme des boîtes noires.

Dans [21] nous avons proposé un état de l'art des réductions entre problèmes qui découlent de la technique ainsi que plusieurs améliorations de pré-conditionneurs. Les matrices de pré-conditionnement sont diagonales, structurées (en particulier Toeplitz), basées sur des réseaux de permutations (papillon ou Beneš) ou creuses. Elles conduisent à des coûts de produits matrice × vecteur en $\mathrm{E}'(n) = \mathrm{E}(n) + \tilde{O}(n \log n)$ ou en $\mathrm{E}'(n) = \mathrm{E}(n) + \tilde{O}\left(n \log^2 n\right)$ selon la taille du corps de base K. Les complexités (théorèmes 6.1.1 et 6.1.2) ne sont donc que peu modifiées asymptotiquement. Le pré-conditionnement est probabiliste en ce sens que $A' = C_g A C_d$ vérifie la propriété cible avec une bonne probabilité.

Pour une matrice inversible, on peut facilement assurer que le polynôme caractéristique soit égal au polynôme minimal. Il s'agit d'une propriété de cyclicité.

Théorème 6.2.1. *Soit $A \in \mathsf{K}^{n\times n}$, S un sous-ensemble fini de K et D une matrice diagonale aléatoire aux coefficients choisis uniformément et indépendamment dans S. Le sous-espace stable associé aux valeurs propres non nulles de DA est cyclique avec une probabilité au moins $1 - n(n-1)/|\mathsf{S}|$ [21, Theorem 4.2].*

Le déterminant est modifié mais on sait dans quel facteur. L'algorithme du paragraphe 4.2 par exemple consiste à calculer le déterminant de $A' = DA$ puis à récupérer $\det A = (\det A')/(\det D)$. Un pré-conditionnement ne modifiant pas le déterminant est cependant possible [121, Theorem 3.2]. Si le corps K n'a pas assez d'éléments pour appliquer le théorème 6.2.1 directement une solution est de passer par une extension algébrique.

Nous verrons au paragraphe suivant l'utilisation du pré-conditionnement pour la résolution de systèmes linéaires généraux. Une autre application est pour le calcul du rang de A. Ce calcul se réduit au calcul du polynôme caractéristique [95, 74]. En effet, si ce dernier est égal à $x^k f(x)$ et que le polynôme minimal est $xf(x)$ avec $f(0) \neq 0$ alors [74] :

$$\mathrm{rang} A = n - k.$$

Cette propriété est assurée en prenant $A' = D_1 A^T D_2 A$ pour D_1 et D_2 deux matrices diagonales [41]. On peut aussi prendre des matrices Toeplitz comme le montre le théorème suivant.

Théorème 6.2.2. *Soit $A \in \mathsf{K}^{n\times n}$ de rang r et S un sous-ensemble fini de K. Soient U, V et W trois matrices Toeplitz aux coefficients aléatoires choisis uniformément et indépendamment dans S ; U est choisie triangulaire supérieure ; V et W sont choisies triangulaires inférieures à diagonales unités. La matrice $VAWU$ est de rang r et son polynôme caractéristique est $f(x)x^{n-r}$ où f est sans carré et tel que $f(0) \neq 0$ avec une probabilité au moins $1 - (3n^2 + n)/|\mathsf{S}|$ [21, Theorem 5.3].*

Un pré-conditionnement Toeplitz est certainement plus coûteux qu'un pré-conditionnement diagonal. Il peut cependant permettre de préserver des propriétés de structure comme par exemple le rang de déplacement.

De même que plus haut, si le corps a trop peu d'éléments pour utiliser la borne sur la probabilité de succès, il est possible de passer par une extension algébrique. Puisqu'il faut au moins de l'ordre de n^2 éléments dans S, l'extension aura un degré en $O(\log n)$. Et puisque, d'autre part, le produit d'une matrice Toeplitz par un vecteur se fait en $\tilde{O}(n \log n)$ [12, §2.5], la complexité de la nouvelle boîte noire est $\mathrm{E}'(n) = \mathrm{E}(n) + \tilde{O}\left(n \log^2 n\right)$ au pire. Une alternative à l'utilisation d'une extension algébrique est d'utiliser une matrice creuse comme pré-conditionneur. Pour un corps quelconque, les résultats connus conduisent cependant à des matrices creuses de poids de Hamming en $O(n \log^2 n)$ et donc aussi à $\mathrm{E}'(n) = \mathrm{E}(n) + \tilde{O}\left(n \log^2 n\right)$ [134, 21]. Du théorème 6.1.1 joint à ces pré-conditionnements on déduit un calcul du rang.

Théorème 6.2.3. *Le rang de A dans $\mathsf{K}^{n\times n}$ se calcule par un algorithme probabiliste Monte Carlo en $O(n\mathrm{E}(n)) + \tilde{O}\left(n^2 \log^2 n\right)$ opérations dans K [134, 74].*

Quitte à pré-conditionner on vient donc de voir une réduction du calcul du rang à celui du polynôme minimal. On peut remarquer que cela vaut aussi pour les méthodes par blocs et donc que les exposants du théorème 3.2.2 (sans division) et du théorème 4.1.4 (sur $\mathbb{Z}$) s'appliquent. Se basant sur le calcul du polynôme minimal, l'algorithme ci-dessus est de type Monte Carlo. On a pu néanmoins produire un certificat de la solution sur $\mathbb{R}$ ou sur $\mathbb{C}$.

Théorème 6.2.4. *Le rang de A dans $K^{n\times n}$ pour $\mathsf{K} = \mathbb{R}$ ou $\mathbb{C}$ se calcule par un algorithme probabiliste Las Vegas en $O(n\mathrm{E}(n) + n^2)$ opérations dans K [108].*

La caractéristique zéro permet ce certificat en assurant la non-existence de vecteurs u isotropes c'est-à-dire tels que $\langle u, u\rangle = 0$.

6.3 Systèmes linéaires : le cas général

Deux cas de la résolution de systèmes linéaires ont déjà été traités. On a vu le calcul d'un vecteur z du noyau donc tel que $Az = 0$ au paragraphe 2.3. La résolution de $Az = b$ pour A inversible a été abordée en début de chapitre avec le polynôme minimal scalaire. Le cas général avec A éventuellement singulière, en utilisant l'algorithme par blocs, repose sur un pré-conditionnement de la matrice [21, §2]. On commence par traiter le cas d'un vecteur b *a priori* dans l'image de A, c'est-à-dire le cas où l'on sait qu'une solution existe. On verra ensuite comment cela se complète par un test de non-existence d'une solution.

Pour $b \neq 0$, on se ramène à la situation où l'on peut diviser par le terme constant du polynôme minimal de b comme dans (6.1). Ce terme constant est non nul pour tout vecteur dans l'image si la forme normale de Jordan de A n'a pas de bloc nilpotent non trivial (A est diagonalisable par rapport à la valeur propre zéro) [128]. La propriété est en particulier induite par celle exigée plus haut pour le rang (théorème 6.2.2) ou peut s'assurer de manière spécifique [21, Corollary 7.3]. De là nous avons proposé de résoudre le système à partir du polynôme minimal $F_X^{A,Z} \in \mathsf{K}[x]^{m\times m}$ avec $Z = [b, AY]$ pour X et Y aléatoires [128]. Les projections sont suffisamment génériques pour que tout fonctionne comme au paragraphe 2.5 avec notamment un polynôme minimal de petit degré $d = n/(m-1)$. Le pré-conditionnement associé à ce choix de Z assure de plus que la matrice terme constant de $F_X^{A,Z}$ est inversible et peut être prise égale à l'identité. Puisque $F_X^{A,Z}$ est polynôme générateur de la suite $XA^i[b, AY]$, sa première colonne mène à une relation

$$b + (\sum_{i=1}^{d} \alpha_i A^i b) + (\sum_{i=0}^{d} \sum_{j=1}^{m-1} \beta_{i,j} A^{i+1} y_j) = 0 \tag{6.4}$$

où les y_j sont les colonnes de $Y \in \mathsf{K}^{n\times(m-1)}$ pour certains éléments α_i et $\beta_{i,j}$ dans K (comparer à (2.7)). L'identité (6.4) généralise (6.1) et donne bien z tel que $Az = b$. Une méthode de résolution un peu moins générale se ramène à un système inversible mais requiert d'abord le rang de la matrice (voir [134, 74, 41] dans le cas scalaire et [67] avec blocs).

La résolution de $Az = b$ pour A quelconque utilise donc un pré-conditionnement et paraît plus difficile que le calcul d'un vecteur du noyau (ou que la résolution avec A inversible) qui n'en nécessite pas. Ceci se perçoit mieux grâce à l'équivalence suivante.

Théorème 6.3.1. *Le problème de résoudre un système linéaire (pour une matrice générale) avec second membre dans l'image est équivalent au problème de calculer un vecteur aléatoire du noyau. C'est-à-dire qu'une résolution donne l'autre avec un surcoût en $O(n)$ [74, 71].*

Le passage par (6.4) pour résoudre le système ne doit bien sûr pas fonctionner si b n'est pas dans l'image. L'assertion qui devient fausse est que la matrice terme constant de $F_X^{A,Z}$ n'est plus forcément inversible. On peut donc détecter avec une bonne probabilité qu'un problème se pose. Mais cela se cantonne à un fort doute. Un certificat a été proposé qui consiste à prouver que b n'est pas dans l'image [56]. Il calcule un vecteur aléatoire w dans le noyau de A à gauche, $wA = 0$, et certifie la non-existence d'une solution si $wb \neq 0$. En effet, à ce moment là, $wAz = wb$ et donc $Az = b$ sont impossibles.

En regroupant tous les résultats, on obtient finalement des complexités analogues à celles vues pour le rang au théorème 6.2.3.

Le modèle de boîte noire que l'on a considéré, basé sur le produit Au, ne mentionnait pas explicitement l'utilisation de la boîte noire transposée et du produit $A^T v$ ou wA. Puisque c'est souvent raisonnable en pratique, on s'autorise cependant l'opération pour le pré-conditionnement $A' =$

$D_1 A^T D_2 A$ page 44 ou pour le certificat de non-existence d'une solution ci-dessus. Plus théoriquement on pourra se reporter aux commentaires de Kaltofen et à sa question ouverte [68, §6]. Dans le cas du certificat, on ne sait pas s'il est possible d'éviter la transposition dans le même ordre de complexité [21, §2].

6.4 Calcul du polynôme caractéristique

Par construction, l'approche Krylov / Lanczos calcule le polynôme minimal. On vient aussi de voir que, grâce aux pré-conditionnements, on pouvait calculer le rang et résoudre des systèmes linéaires en $O(n)$ produits matrice $\times$ vecteur plus $\tilde{O}\left(n^2 \log^2 n\right)$ opérations additionnelles. On ne connaît pas de solution analogue pour le calcul du polynôme caractéristique.

Problème 6.4.1. *Sur un corps abstrait* K*, calculer le polynôme caractéristique* χ_A *de* $A \in \mathsf{K}^{n \times n}$ *en temps* $\tilde{O}\left(n\mathrm{E}(n) + n^2\right)$ *et en mémoire* $\tilde{O}(n)$ *[68, Open Problem 3].*

La question peut aussi être vue sous l'angle du calcul de toutes les valeurs propres de la matrice [101]. Dans le pire cas sous le modèle boîte noire, la meilleure complexité connue n'était pas meilleure que pour une matrice générale, donc en $O(n^3)$ ou $\tilde{O}(n^\omega)$ (théorème 1.1.1). Des bornes plus fines existent seulement pour des classes de matrices particulières comme les matrices structurées ou les matrices creuses définies par des graphes $s(n)$-séparables [102, 101, 106]. De même, Eberly a proposé un algorithme adaptatif en le nombre de blocs ϕ de la forme normale de Frobenius de A (voir §1.7). Il obtient un coût en $O(n\mathrm{E}(n) + \phi n^2)$ qui donne cependant toujours un terme en $O(n^3)$ dans le pire cas $\phi = O(n)$ [37, 38].

Nous avons proposé une méthode qui réduit le nombre de produits matrice $\times$ vecteur dans le cas général et qui ne fait pas intervenir de terme en $O(n^3)$. Elle permet de passer en deçà d'une complexité globale en $O(n^3)$ — ou en $O(n^\omega)$ avec produit rapide — dès que $\mathrm{E}(n) = o(n^2)$, soit quand $\mathrm{E}(n) = \tilde{O}(n)$ par exemple. La borne cible du problème 6.4.1 n'est pas pour autant atteinte. Le résultat correspond à l'objectif intermédiaire donné en introduction au chapitre.

Notre démarche ici est à rapprocher de celle du chapitre 4 sur $\mathbb{Z}$. On y a vu en effet que calculer le plus grand facteur invariant était "plus facile" que de calculer le produit de tous les facteurs invariants, c'est-à-dire le déterminant (corollaire 4.1.2). La question se répète puisque le modèle boîte noire calcule le polynôme minimal, le plus grand facteur invariant f_1 de A, mais pas le produit (1.13) :

$$\chi_A(x) = \det(xI - A) = f_1(x) f_2(x) \dots f_\phi(x).$$

Rappelons que les n facteurs invariants de $xI - A$ sont les ϕ polynômes $s_i = f_i$, $1 \leq i \leq \phi$, correspondants aux ϕ blocs de la forme normale de Frobenius de A et les $n - \phi$ polynômes triviaux $s_{\phi+1} = \ldots = s_n = 1$.

Le modèle boîte noire donne moins de latitude que le travail sur des matrices générales. Il s'ensuit que tout le spectre des solutions développées pour le déterminant d'une matrice dense ne s'applique pas pour le calcul du déterminant de $xI - A$ quand A est une boîte noire. La seule solution commune pour l'instant, est d'utiliser la technique des perturbations additives vue au paragraphe 4.1 (initialement développée dans le présent contexte, d'ailleurs) [130]. C'est ce que nous exposons ici brièvement en introduisant une amélioration en la recherche binaire par blocs des facteurs invariants [129].

Pour préserver le modèle boîte noire, il s'impose de choisir des perturbations additives souscrivant au modèle. La proposition 4.1.3 qui perturbait avec des matrices denses se spécialise avec des matrices Toeplitz.

Proposition 6.4.2. *Soit $A \in \mathsf{K}^{n\times n}$ avec $s_1, \ldots, s_n \in \mathsf{K}[x]$ les facteurs invariants de $xI - A$. Soit S un sous-ensemble fini de K et $U \in \mathsf{K}^{n\times k}$ et $V \in \mathsf{K}^{k\times n}$ deux matrices Toeplitz aux coefficients aléatoires choisis uniformément et indépendamment dans S. Les facteurs invariants de $A + UV$ sont*

$$s_{k+1}(x)t_1(x), \ldots, s_n(x)t_{n-k}(x), t_{n-k+1}(x), \ldots, t_n(x)$$

où les $t_i \in \mathsf{K}[x]$, $1 \le i \le n$, sont premiers avec le polynôme caractéristique de A avec une probabilité au moins $1 - (nk + n + 1)/|\mathsf{S}|$.

On retrouve donc le fait que le pgcd du plus grand facteur invariant de A et de celui de $A+UV$ est s_{k+1} quand U et V sont Toeplitz de rang k. Comme quand il s'est agi du pré-conditionnement, la complexité de la nouvelle boîte noire $A + UV$ est $\mathrm{E}'(n) = \mathrm{E}(n) + \tilde{O}\left(n \log^2 n\right)$.

En plus de la perturbation structurée, une autre différence avec la méthode vue sur $\mathbb{Z}$ est que l'on peut calculer ici plusieurs facteurs invariants simultanément. Sur $\mathbb{Z}$, les facteurs étaient calculés séparément par résolutions de systèmes linéaires. En suivant le théorème 2.1.1, pour un choix m de taille de blocs on peut considérer le polynôme minimal $F_X^{A+UV,Y} \in \mathsf{K}[x]^{m\times m}$. Pour X et Y aléatoires, on sait d'après la proposition précédente que ses m facteurs invariants sont

$$s_{k+1}(x)t_1(x), \ldots, s_{k+m}(x)t_m(x). \tag{6.5}$$

Cela donne le lemme suivant.

Lemme 6.4.3. *Pour k fixé quelconque, on peut calculer $s_{k+1}, \ldots, s_{k+m}$ par un algorithme Monte Carlo en $O(n\mathrm{E}(n)) + \tilde{O}\left(n^2 + m^{\omega-1}n\right) + \mathsf{MM}(n, m)$ opérations dans K où la fonction MM donne le coût du produit de matrices rectangulaires (cf théorème 3.2.1) .*

Démonstration. On pré-calcule $s_1 = f_1$ par le théorème 6.1.1. Pour obtenir les polynômes de (6.5) on calcule ensuite $F_X^{A+UV,Y}$. Le théorème 2.5.1 nous dit que ce polynôme matriciel est de degré n/m et qu'il se calcule en temps $\tilde{O}\left(m^{\omega-1}n\right)$ à partir des $2d$ premiers termes de la suite $X(A+UV)^iY$. Ces termes s'obtiennent eux-mêmes en $O(n\mathrm{E}(n))+\tilde{O}\left(n^2\right)+\mathsf{MM}(n, m)$ opérations. Par le théorème 5.2.1 on en déduit les facteurs invariants (6.5) de $F_X^{A+UV,Y}$ en temps $\tilde{O}\left(m^{\omega-1}n\right)$ et finalement $s_{k+i} = \mathrm{pgcd}(s_{k+i}t_i, s_1)$, $1 \le i \le m$. □

L'algorithme de calcul de la forme normale de Smith de $xI - A$ et donc du polynôme caractéristique de A est alors une recherche binaire par blocs analogue à celle de la page 28. Il faut appliquer de l'ordre de μ_m fois le lemme 6.4.3 où μ_m est le nombre de changements de degrés possibles entre deux produits

$$s_{i+1}s_{i+2}\ldots s_{i+m} \text{ et } s_{j+1}s_{j+2}\ldots s_{j+m},\ i + m < j + 1,$$

de m facteurs invariants consécutifs. On vérifie que la divisibilité des facteurs invariants implique que $\mu_m = O(\sqrt{n/m})$. Le coût global du calcul correspond donc à $\tilde{O}(\sqrt{n/m})$ utilisations du lemme 6.4.3. En particulier, pour des matrices assez creuses *i.e.* avec $\tilde{O}(n)$ éléments non nuls ou des matrices assez structurées, la valeur optimale pour m permet de conclure comme suit.

Théorème 6.4.4. *On suppose que la boîte noire A est telle que $\mathrm{E}(n) = \tilde{O}(n)$ et que α est la constante du produit de matrices rectangulaires du théorème 3.2.1. Les facteurs invariants de A (et donc son polynôme caractéristique) se calculent en $\tilde{O}\left(n^{2+(1-\alpha)/2}\right) = O(n^{2.36})$ opérations dans K dont $\tilde{O}\left(n^{1+(1-\alpha)/2}\right)$ applications de la noîte noire. En produits usuel de matrices, pour $m = 1$, on retrouve $\tilde{O}\left(n^{1+1/2}\mathrm{E}(n) + n^{2+1/2}\right)$ opérations dans K [130].*

Pour un produit matrice × vecteur suffisamment économique, le théorème vient donc d'établir un exposant plus faible que l'exposant du produit de matrices.

6.5 Prolongements

Jusque-là nous avons été concernés par des boîtes noires sur un corps K abstrait. Le même type de questions peut se poser pour des matrices boîtes noires sur $\mathbb{Z}$ par exemple.

La forme normale de Smith de matrice creuses sur $\mathbb{Z}$ a plus particulièrement été étudiée. Une technique de base est celle de Giesbrecht (voir avant le corollaire 4.3.1 page 30) qui réduit le calcul de la forme à celui de plusieurs polynômes minimaux [55]. Toujours à partir de coefficients de polynômes minimaux, avec J.G. Dumas et B.D. Saunders, nous avons développé une méthode s'appuyant sur des calculs de rangs [35]. À partir du calcul du plus grand facteur invariant, Giesbrecht a aussi développé une méthode pour la résolution de systèmes diophantiens creux [54] (l'étude est complétée dans le cas dense par [93]). Le cas de la forme normale de Frobenius a été traité par Storjohann. Son idée d'appliquer la remontée de Hensel au calcul de cette forme sur $\mathbb{Z}$ (nous l'avons reprise pour des matrices générales page 30) a été initialement développée pour des matrices creuses [115].

Conclusion

En complexité algébrique sur un corps, on peut dire que l'algèbre linéaire de base, comparée au produit de matrices en n^ω, est assez bien comprise (résolution de systèmes mise à part). Sous des modèles de calcul généraux, de nombreuses équivalences entre problèmes sont en effet connues. En complexité algébrique sur $\mathsf{K}[x]$ ou en complexité binaire sur $\mathbb{Z}$, la scène est moins éclairée. Disons qu'il manque à la fois un ou des exposants cibles ainsi que des réductions entre problèmes.

Les progrès récents, dont ceux présentés dans ce mémoire, nous apprennent sur les exposants. Il est nouveau de savoir que le déterminant, la forme normale de Smith et l'inverse d'une matrice polynomiale sur $\mathsf{K}[x]$ sont ou seraient accessibles en *grosso modo* d produits de matrices sur K. On peut à juste titre espérer que ces problèmes pour une matrice entière puissent aussi se résoudre en approximativement $\log \|A\|$ produits de matrices. En notant l la longueur d ou $\log \|A\|$ des coefficients en entrée, une première question est donc de savoir quels sont les problèmes qui peuvent être résolus en temps $\tilde{O}(n^\omega l)$ plus la longueur de la sortie (algébrique ou binaire) ? Les problèmes les plus difficiles sont sur $\mathbb{Z}$ et sont ceux dont la sortie est une matrice. Il reste par exemple à calculer le polynôme caractéristique. Il reste à donner des algorithmes généraux pour l'inversion et le calcul des formes normales (Popov et Hermite) avec transformations associées. Parallèlement à cela, le procédé d'élimination de Strassen amène des séries formelles et un lien peut-être fortuit avec les matrices polynomiales. La question de calculer le déterminant sans division en temps $\tilde{O}(n^\omega)$ reste intriguante.

Les avancées notamment autour du déterminant révèlent le manque de réductions entre problèmes puisque ces avancées ne se transportent pour l'instant qu'à peu d'autres situations. Indépendamment sur $\mathsf{K}[x]$ ou $\mathbb{Z}$, une autre question est donc de dégager des réductions et des équivalences entre problèmes.

Nous sommes curieux des mêmes réflexions pour les matrices structurées voire les boîtes noires.

Notre exposé théorique est simplifié à plus d'un titre. En appuyant sur l'exposant de n on oublie d'étudier le rôle des facteurs logarithmiques. On oublie aussi que les algorithmes sont indisctinctement déterministes ou probabilistes de type Las Vegas voire Monte Carlo. On oublie enfin que les machines peuvent être limitées en mémoire.

Les méthodes de Gauss, de Krylov et de Padé jouent des rôles très proches ou complémentaires dans ces problèmes matriciels polynomiaux et binaires. Il y a quoi qu'il en soit matière à préciser notre compréhension.

Annexe

A — Démonstration de la proposition 1.9.3 page 11

Le coût du calcul et l'identité (1.15) ont été donnés par Storjohann [116, §9]. En renversant l'ordre des coefficients, cette identité est une division euclidienne de matrices polynomiales. Quant aux propriétés de la nouvelle fraction, montrons d'abord qu'elle est strictement propre. On a

$$x^h R(x)Q(x)^{-1} = P(x)Q(x)^{-1} - \left(P(x)Q(x)^{-1} \bmod x^h\right). \tag{6.6}$$

En divisant P par Q à droite on peut obtenir (voir [64, Theorem 6.3-15])

$$P(x)Q(x)^{-1} = N(x) + T(x)Q(x)^{-1}$$

avec N une matrice polynomiale et TQ^{-1} strictement propre. Si A^* est la matrice adjointe de A, on a $PQ^{-1} = A^{-1}B = A^*B/(\det A)$ et puisque TQ^{-1} est strictement propre, $\deg N \leq \deg A^*B \leq (n-1)d + \deg B < h$. La dernière inégalité est donnée par hypothèse. Donc le terme de droite de (6.6) multiplié par x^{-h} sera strictement propre, c'est ce qu'on voulait montrer.

Concernant le pgcd de P et de Q comparé à celui de R et de Q, en notant $M(x) = P(x)Q(x)^{-1} \bmod x^h$ on a

$$\begin{bmatrix} x^h R(x) \\ Q(x) \end{bmatrix} = \begin{bmatrix} I & -M \\ 0 & I \end{bmatrix} \begin{bmatrix} P(x) \\ Q(x) \end{bmatrix}$$

donc $\operatorname{pgcd}(x^h R, Q) = \operatorname{pgcd}(P, Q)$ (voir (1.11)). Démontrons donc que $\operatorname{pgcd}(x^h R, Q) = \operatorname{pgcd}(R, Q)$. Soit $G = \operatorname{pgcd}(R, Q) \in \mathsf{K}[x]^{m\times m}$ avec $R = \bar{R}G$ et $Q = \bar{Q}G$ et $G' = \operatorname{pgcd}(x^h R, Q) \in \mathsf{K}[x]^{m\times m}$. D'après le théorème 1.6.3 on a $G' = VG$. Donc $V \in \mathsf{K}[x]^{m\times m}$ est diviseur commun à droite de $x^h\bar{R}$ et de $\bar{Q}$ avec $\bar{R}$ et de $\bar{Q}$ premières entre elles. Si V n'était pas unimodulaire il existerait $x_0 \in \bar{\mathsf{K}}$ tel que $\det V(x_0) = 0$ et u non nul dans $\bar{\mathsf{K}}^m$ tel que $V(x_0)u = 0$. On aurait donc $(x_0^h\bar{R}(x_0))u = 0$ et $\bar{Q}(x_0)u = 0$. Le corollaire 1.6.4 pour $\bar{R}$ et de $\bar{Q}$, donnerait alors $x_0 = 0$ et mènerait à une contradiction puisque par hypothèse $\det Q(0) \neq 0$. Puisque V est unimodulaire et par unicité du pgcd (forme normale) alors $G' = G$ et $\operatorname{pgcd}(x^h R, Q) = \operatorname{pgcd}(R, Q) = \operatorname{pgcd}(P, Q)$.

B — Démonstration du théorème 2.1.1 page 14

Lemme B. *Soit $M \in \mathsf{K}[x]^{\mu\times\mu}$ non singulière et soit $U \in \mathsf{K}[x]^{\mu\times\mu}$ unimodulaire telle que*

$$MU = \begin{bmatrix} H & H_{12} \\ 0 & H_{22} \end{bmatrix} \tag{6.7}$$

avec H une matrice carrée, alors le i-ème facteur invariant de H divise le i-ème facteur invariant de M.

Démonstration. On peut réécrire (6.7) comme

$$MU = \begin{bmatrix} I & H_{12} \\ 0 & H_{22} \end{bmatrix} \begin{bmatrix} H & \\ & I \end{bmatrix}.$$

Puisque les facteurs invariants du produit de deux matrices non singulières divisent les facteurs invariants du produit, les plus grands facteurs invariants de $\mathrm{diag}(I, H)$ qui sont ceux de H divisent ceux correspondants de MU donc de M. □

Démonstration du théorème. On regroupe essentiellement plusieurs résultats de Kailath [64, §6.4.2]. Remarquons d'abord que l'on peut se limiter au cas où Y est de plein rang en colonnes. Si Y est de rang $r < m$, on peut en effet introduire une transformation inversible $Q \in \mathsf{K}^{m\times m}$ telle que $YQ = [Y_1, 0]$ avec $Y_1 \in \mathsf{K}^{n\times r}$. De là, si

$$X^T(xI - A)^{-1}Y_1 = N_1(x)D_1(x)^{-1} \tag{6.8}$$

est une description irréductible alors

$$X(xI - A)^{-1}Y = \begin{bmatrix} N_1(x) & 0 \end{bmatrix} \begin{bmatrix} D_1(x) & \\ & I \end{bmatrix}^{-1} Q^{-1}$$

est une description irréductible de $X(xI - A)^{-1}Y$. Si bien que, puisque les facteurs invariants non triviaux de $Q\ \mathrm{diag}(D_1, I)$ sont ceux de D_1, on peut effectivement se limiter à étudier (6.8) avec Y_1 de plein rang.

Montrons ensuite la relation de divisibilité entre les facteurs invariants de $F_X^{A,Y}$ et ceux de A. On construit pour cela une description particulière de $X(xI - A)^{-1}Y$. Soit $Y_c \in \mathsf{K}^{n\times(n-m)}$ choisie telle que $T = [Y, Y_c]$ soit inversible dans $\mathsf{K}^{n\times n}$. Soit aussi $H_Y(x) \in \mathsf{K}[x]^{m\times m}$ définie à partir d'une triangularisation unimodulaire de $T^{-1}(xI - A)$, c'est-à-dire que :

$$T^{-1}(xI - A)U(x) = \begin{bmatrix} H_Y(x) & H_{12}(x) \\ 0 & H_{22}(x) \end{bmatrix} \tag{6.9}$$

avec U unimodulaire. Si V est la matrice formée des m premières colonnes de U, on a donc les descriptions

$$(xI - A)^{-1}Y = V(x)H_Y(x)^{-1}$$

et

$$X(xI - A)^{-1}Y = (XV(x))\, H_Y(x)^{-1}. \tag{6.10}$$

Par cette identité, le théorème 1.6.5 et le lemme 2.1.2, on sait que H_Y est multiple de $F_X^{A,Y}$. En utilisant l'argument sur le produit de matrices (voir la preuve du lemme B) on sait donc que les facteurs invariants de $F_X^{A,Y}$ divisent ceux de H_Y. Ces derniers divisent eux-mêmes — d'après la construction (6.9) de H_Y et le lemme B — les facteurs invariants de $xI - A$. Cela démontre la première assertion.

Pour la deuxième assertion et l'existence de W et de Z on va chercher à obtenir, dans l'application du lemme B et du théorème 1.6.5, non seulement la divisibilité mais aussi la coïncidence des facteurs invariants.

Concernant le lemme B, montrons que l'on peut choisir T et donc Z (une matrice Y particulière) telles que les facteurs invariants de H_Y soient exactement les f_i. Soit C la forme normale de Frobenius de A. La matrice C est diagonale par blocs et les polynômes caractéristiques de ses

blocs sont les f_i, $1 \leq i \leq \phi$ (voir §1.7). On note d_i le degré de f_i. Pour des raisons techniques, on prend ici des blocs compagnons "lignes" : $C'_{f_i} = J^{-1}C_{f_i}^T J$ si J est la matrice anti-diagonale de 1 et si C_{f_i} est un bloc compagnon "colonne". On définit $\Psi \in \mathsf{K}[x]^{n\times\phi}$ par :

$$\Psi^T(x) = \mathrm{diag}([x^{d_1-1}, x^{d_1-2}, \ldots, 1], \ldots, [x^{d_\phi-1}, x^{d_\phi-2}, \ldots, 1]) \in \mathsf{K}^{\phi\times n}.$$

Il est alors possible de vérifier que $(xI - C)\Psi = T_1[\mathrm{diag}(f_1, f_2, \ldots, f_\phi), 0]^T$ où T_1 est une permutation des lignes. Puisque l'on peut extraire de Ψ la matrice identité $\phi \times \phi$ on sait que Ψ peut être complétée en matrice unimodulaire $U_1 = [\Psi, \Phi]$. En notant $S_Y = \mathrm{diag}(f_1, f_2, \ldots, f_\phi)$ on a donc :

$$T_1^{-1}(xI - C)U_1(x) = \begin{bmatrix} S_Y(x) & S_{12}(x) \\ 0 & S_{22}(x) \end{bmatrix}.$$

Les facteurs invariants non triviaux des deux côtés de cette égalité sont donnés par S_Y, il existe donc une matrice unimodulaire U_2 telle que

$$T_1^{-1}(xI - C)U_1(x)U_2(x) = \begin{bmatrix} S_Y(x) & 0 \\ 0 & I \end{bmatrix}, \quad U_2(x) = \begin{bmatrix} I & U_{12}(x) \\ 0 & U_{22}(x) \end{bmatrix}.$$

Avec T_2 donnée par $C = T_2^{-1}AT_2$ (C est semblable à A) cela donne aussi

$$T_1^{-1}T_2^{-1}(xI - A)T_2U_1(x)U_2(x) = \begin{bmatrix} S_Y(x) & 0 \\ 0 & I \end{bmatrix}.$$

En comparant cette dernière égalité avec (6.9) on voit qu'en choisissant Z d'après $T = T_2T_1$ et en prenant $U = T_2U_1U_2$ on obtient que les facteurs invariants de H_Y sont comme voulu $f_1, \ldots, f_m$ quand $m \leq \phi$ et $f_1, \ldots, f_\phi, 1, \ldots, 1$ quand $m > \phi$.

Pour terminer il faut montrer — pour utiliser le théorème 1.6.5 et le lemme 2.1.2 — qu'il existe un choix de W (une matrice X particulière) approprié qui garde les même facteurs invariants au dénominateur d'une description irréductible $W^T(xI - A)^{-1}Z = N(x)D(x)^{-1}$. On regarde ce qui se passe dans (6.10) avec H_Y correspondant au choix de Z, où V est donnée par les m premières colonnes de U, et on discute suivant la valeur de $\min\{l, m, \phi\}$. Si $m \leq \phi$, on peut extraire des m premières colonnes de U_1U_2 la matrice identité $m \times m$, on peut donc trouver $W \in \mathsf{K}^{l\times n}$ telle que

$$WV(x) = WT_2U_1(x)U_2(x)\begin{bmatrix} I_m \\ 0 \end{bmatrix} = \begin{bmatrix} I_{\min\{l,m\}} & 0 \\ 0 & 0 \end{bmatrix} \in \mathsf{K}^{l\times m}$$

où le nombre de lignes nulles et le nombre de colonnes nulles dépendent de $\min\{l, m\}$. Si $l \geq m$ alors WV contient l'identité I_m et est forcément première avec $H_Y = \mathrm{diag}(f_1, \ldots, f_m)$. On peut donc prendre cette dernière matrice au dénominateur d'une description irréductible, c'est ce qu'il fallait démontrer. Si $l \leq m$ alors il existe une matrice unimodulaire R telle que

$$R(x)\begin{bmatrix} WV(x) \\ H_Y \end{bmatrix} = R(x)\begin{bmatrix} I_l & 0 \\ \mathrm{diag}(f_1(x), \ldots, f_m(x)) \end{bmatrix} = \begin{bmatrix} I_l & 0 \\ 0 & \mathrm{diag}(f_{l+1}(x), \ldots, f_m(x)) \\ 0 & 0 \end{bmatrix}$$

ce qui montre que $\mathrm{diag}(I_l, f_{l+1}, \ldots, f_m)$ est un pgcd à droite de $WV(x)$ et de $H_Y(x)$. En simplifiant numérateur et dénominateur par ce pgcd, on montre donc que l'on peut prendre $\mathrm{diag}(f_1, \ldots, f_l, I_{m-l})$ ce qui établit aussi le résultat. De façon analogue, quand $m > \phi$ on peut trouver $W \in \mathsf{K}^{l\times n}$ telle que

$$WV(x) = WT_2U_1(x)U_2(x)\begin{bmatrix} I_m \\ 0 \end{bmatrix} = \begin{bmatrix} I_{\min\{l,\phi\}} & 0 & V_{13}(x) \\ 0 & 0 & V_{23}(x) \end{bmatrix} \in \mathsf{K}^{l\times m}.$$

Donc si $l \leq \phi$, un pgcd à droite de $WV(x)$ et de $H_Y(x)$ est $\mathrm{diag}(I_l, f_{l+1}, \ldots, f_\phi)$ et on peut prendre $\mathrm{diag}(f_1, \ldots, f_l, I_{m-l})$ comme dénominateur réduit. Si $l > \phi$, $WV(x)$ et $H_Y(x)$ sont premières entre elles et on prend $\mathrm{diag}(f_1, \ldots, f_\phi)$ au dénominateur. Ceci conclut la preuve de la deuxième assertion.

Concernant la maximalité du degré total ainsi obtenu, pour m fixé, les facteurs invariants $f_1, \ldots, f_{\min\{m,\phi\}}, 1 \ldots, 1$ de $F^{A,Z}$ ont les degrés les plus grands possibles puisque ce sont les m plus grands facteurs invariants de A. De surcroît, pour l aussi fixé, la construction de W correspond à un pgcd à droite, pour XV et H_Y, de degré de déterminant aussi petit que possible. Le dénominateur réduit obtenu après simplification a donc un degré de déterminant maximal, c'est-à-dire la somme des degrés des $\min\{l, m, \phi\}$ plus grands facteurs invariants.

C — Démonstration de la proposition 2.2.1 page 15

La preuve de Wiedemann pourrait se généraliser au cas matriciel. On procède un peu différemment en liaison avec ce que l'on a déjà construit. Puisque $F^{A,Y}$ engendre $\{XA^iY\}_{i\geq 0}, X(xI - A)^{-1}Y$ admet une description de dénominateur $F^{A,Y}$:

$$H(x) = X(xI - A)^{-1}Y = N(x)F^{A,Y}(x)^{-1} \in \mathsf{K}(x)^{l\times m}.$$

Montrons que l'on peut prendre

$$\zeta(X) = X(xI - A)^{-1}YF^{A,Y}(x) = N(x) \in \mathsf{K}[x]^{l\times m}.$$

Puisque $H(x)$ est strictement propre (tend vers 0 en l'infini), $N(x)$ appartient bien à $\mathcal{M}_{A,Y}$. Donc ζ est une application linéaire. Montrons qu'elle est surjective. Puisque $F^{A,Y}$ est colonne réduite, $N(x)$ détermine $H(x) = \sum H_i/x^{i+1}$. Quitte à considérer la restriction A_Y de A au sous-espace stable $\langle Y, AY, A^2Y, \ldots\rangle$, on suppose que $\langle Y, AY, A^2Y, \ldots\rangle = \mathsf{K}^n$. Les identités $H_i = XA^iY$ déterminent X complètement. Il reste à démontrer que ζ donne bien l'équivalence avec la primalité des matrices. C'est le cas d'après le lemme 2.1.2 avec $N(x) = \zeta(X)$ et $D(x) = F^{A,Y}(x)$.

D1 — Démonstration du théoreme 2.5.2 page 19

Les assertions peuvent être montrées par récurrence. Pour $i = 0$, puisque H_0 est inversible, $M^{(0)}$ satisfait au point I) du théorème. Par définition $F^{(0)} = I$ et en commençant à $i = 1$, $S^{(1)} = I$. On suppose les propriétés vraies pour $i - 1$. Avec (2.14) on a,

$$Q^{(i)} = \tilde{Q}^{(i)}x + \bar{Q}^{(i)} = \left(M^{(i-1)}_{2d-i}\right)^{-1} M^{(i-2)}_{2d-i+1}x + \bar{Q}^{(i)} \ ,$$

$\tilde{Q}^{(i)}$ est inversible d'après I) aux étapes précédentes et $\bar{Q}^{(i)}$ est dans $\mathsf{K}^{m\times m}$. La matrice de tête de $F^{(i)}$ est

$$F^{(i)}_i = -F^{(i-1)}_{i-1}\tilde{Q}^{(i)}$$

donc $F^{(i)}$ satisfait à II) pour $0 \leq i - 1 \leq d - 1$. Le même argument peut être utilisé pour $S^{(i)}$ $(1 \leq i-1 \leq d-1)$. Par construction, $M^{(i)}$ est de degré moins que $2d-1-i$ donc, pour les matrices

coefficients à droite dans (2.16) on sait que pour $0 \leq i-1 \leq d-2$:

$$\begin{bmatrix} H_0 & H_1 & \dots & H_i \\ H_1 & H_2 & \dots & H_{i+1} \\ \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ H_i & H_{i+1} & \dots & H_{2i} \end{bmatrix} \begin{bmatrix} F_0^{(i)} \\ F_1^{(i)} \\ \vdots \\ \vdots \\ F_i^{(i)} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ M_{2d-1-i}^{(i)} \end{bmatrix}.$$

Par hypothèse sur H et puisque l'on a prouvé que $F_i^{(i)}$ est inversible, le terme de droite de l'équation précédente est de rang m et donc $M_{2d-1-i}^{(i)}$ est inversible. L'identité (2.16) pour $i = d$ donne aussi (2.12), c'est-à-dire que $F^{(d)}$ est un polynôme minimal pour $\{H_i\}_{0 \leq i \leq 2d-1}$ dont toutes les colonnes sont de degré d exactement. Il est minimal car le fait que $\mathcal{M}$ soit inversible implique qu'un vecteur générateur (une colonne) ne peut être de degré inférieur à d.

D2 — Démonstration du lemme 2.5.5 page 20

On se fonde sur la construction de la preuve de la proposition 6 dans [125]. Si $\rho = \nu - m(d-1)$ on sait que l'on peut trouver une matrice $Z \in \mathsf{K}^{n \times m}$ telle que

$$\operatorname{rang}\left[Z \mid AZ \mid A^2Z \mid \dots \mid A^{d-2}Z \mid A^{d-1}\left(Z_1 \mid Z_2 \mid \dots \mid Z_\rho\right)\right] = \nu.$$

où Z_j est la j-ème colonne de Z. Il suit que si $K_i(A, \mathcal{Z}) \in \mathsf{K}^{n \times i}$ consiste en les i premières colonnes de $\left[\mathcal{Z}, A\mathcal{Z}, \dots, A^{d-1}\mathcal{Z}\right]$, alors

$$\operatorname{rang} K_i(A, \mathcal{Z}) = i, 1 \leq i \leq \nu. \tag{6.11}$$

Montrons alors que la sous-matrice principale $i \times i$, $\mathcal{M}_i(A, \mathcal{X}, \mathcal{Z})$, de $\mathcal{M}(A, \mathcal{X}, \mathcal{Z})$ est de rang i pour $1 \leq i \leq \nu$. Si $i \leq m$ d'après (6.11) il suffit de voir que l'on peut trouver une matrice $X \in \mathsf{K}^{i \times n}$ telle que $XK_i(A, \mathcal{Z})$ soit de rang i. Pour $i > m$, d'après (6.11) aussi, on peut trouver $X \in \mathsf{K}^{m \times n}$ telle que

$$XK_i(A, \mathcal{Z}) = \begin{bmatrix} 0 & J_m \end{bmatrix} \in \mathsf{K}^{m \times i}$$

où J est la matrice anti-diagonale $m \times m$ de 1. Puique les lignes de $XK_i(A, \mathcal{Z})$ sont les m premières lignes de $\mathcal{M}_i(A, X, \mathcal{Z})$, cette dernière matrice qui est Hankel par blocs a des 1 sur son anti-diagonale et des zéros au-dessus. Elle est donc de rang i.

Bibliographie

[1] J. Abbott, M. Bronstein, and T. Mulders. Fast deterministic computation of determinants of dense matrices. In *International Symposium on Symbolic and Algebraic Computation, Vancouver, BC, Canada*, pages 197–204. ACM Press, Jul 1999. [p. 27]

[2] J. Abdeljaoued and H. Lombardi. *Méthodes matricielles, Introduction à la Complexité Algébrique*. En préparation. [pp. v, 23]

[3] E.H. Bareiss. Computational solution of matrix problems over an integral domain. *J. Inst. Math. Appl.*, 10:68–104, 1972. [p. 9]

[4] W. Baur and V. Strassen. The complexity of partial derivatives. *Theor. Comp. Sc.*, 22:317–330, 1982. [pp. 1, 26]

[5] B. Beckermann and G. Labahn. A uniform approach for the fast computation of matrix-type Padé approximants. *SIAM J. Matrix Anal. Appl.*, 15(3):804–823, July 1994. [pp. 6, 33]

[6] B. Beckermann and G. Labahn. Fraction-free computation of matrix rational interpolants and matrix GCD's. *SIAM J. Matrix Anal. Appl.*, 22(1):114–144, 2000. [p. 6]

[7] B. Beckermann, G. Labahn, and G. Villard. Normal forms for general polynomial matrices. Research Report 2002-1, Laboratoire LIP, ENS Lyon, France, 2002. [pp. 3, 4, 33]

[8] B. Beckermann, G. Labahn, and G. Villard. Shifted normal forms of polynomial matrices. In *International Symposium on Symbolic and Algebraic Computation, Vancouver, Canada*, pages 189–196. ACM Press, Jul. 1999. [pp. 3, 4, 6, 33]

[9] T. Beelen and P.M. Van Dooren. An improved algorithm for the computation of Kronecker's canonical form of a singular pencil. *Linear Algebra and its Applications*, 105:9–65, 1988. [p. 32]

[10] T. Beelen and P.M. Van Dooren. A pencil approach for embedding a polynomial matrix into a unimodular matrix. *SIAM J. Matrix Anal. Appl.*, 9(1):77–89, Jan. 1988. [p. 32]

[11] S.J. Berkowitz. On computing the determinant in small parallel time using a small number of processors. *Inform. Process. Letters*, 18:147–150, 1984. [p. 10]

[12] D. Bini and V.Y. Pan. *Polynomial and matrix computations*. Birkhäuser, 1994. [pp. iv, 1, 9, 44]

[13] R.R. Bitmead and B.D.O. Anderson. Asymptotically fast solution of Toeplitz and related systems of linear equations. *Linear Algebra and its Applications*, 34:103–116, 1980. [p. 6]

[14] R.P. Brent, S. Gao, and G.B. Lauder. Random krylov spaces over finite fields. Manuscript, Oxford University Computing Laboratory, 2001. [p. 15]

[15] R.P. Brent, F.G. Gustavson, and D.Y.Y Yun. Fast solution of Toeplitz systems of equations and computation of Padé approximations. *Journal of Algorithms*, 1:259–295, 1980. [p. 6]

[16] W.S. Brown and J.F. Traub. On Euclid's Algorithm and the Theory of Subresultants. *Journal of the ACM*, 18(4):505–514, 1971. [p. 18]

[17] J. Bunch and J. Hopcroft. Triangular factorization and inversion by fast matrix multiplication. *Math. Comp.*, 28:231–236, 1974. [p. 1]

[18] P. Bürgisser, M. Clausen, and M.A. Shokrollahi. *Algebraic Complexity Theory.* Volume 315, Grundlehren der mathematischen Wissenschaften. Springer-Verlag, 1997. [pp. iv, 1, 9]

[19] S. Cabay and T.P.L. Lam. Congruence techniques for the exact solution of integer systems of linear equations. *ACM Trans, Math. Software*, 3(4):386–397, 1977. [p. 9]

[20] D.G. Cantor and E. Kaltofen. On fast multiplication of polynomials over arbitrary algebras. *Acta Informatica*, 28(7):693–701, 1991. [pp. iv, 20, 26]

[21] L. Chen, W. Eberly, E. Kaltofen, B.D. Saunders, W.J. Turner, and G. Villard. Efficient matrix preconditioners for black box linear algebra. *Linear Algebra and its Applications*, 343-344:119–146, 2002. [pp. 43, 44, 45, 46]

[22] A.L. Chistov. Fast parallel computation of the rank of matrices over a field of arbitrary characteristic. In *Proc. FCT'85*, LNCS 199, pages 63–69. Springer Verlag, 1985. [p. 10]

[23] von Ulrich Christian. Über teilerfremde symmetrische Matrizenpaare. *J. für die reine and angew. Math.*, 229:43–49, 1968. [p. 15]

[24] *Computer Algebra Handbook*, J. Grabmeier, E. Kaltofen, and V. Weispfenning editors, Springer Verlag, Heidelberg, Germany, 2002. [p. iv]

[25] D. Coppersmith. Solving linear equations over GF(2): block Lanczos algorithm. *Linear Algebra and its Applications*, 192:33–60, 1993. [p. 21]

[26] D. Coppersmith. Solving homogeneous linear equations over GF(2) via block Wiedemann algorithm. *Math. Comp.*, 62(205):333–350, 1994. [pp. 6, 13, 15, 16]

[27] D. Coppersmith. Rectangular matrix multiplication revisited. *J. of Complexity*, 13:42–49, 1997. [pp. 1, 25]

[28] D. Coppersmith and S. Winograd. Matrix multiplication via arithmetic progressions. *J. of Symbolic Computations*, 9(3):251–280, 1990. [p. 1]

[29] L. Csanky. Fast parallel matrix inversion algorithms. *SIAM J. Comput.*, 5(4):618–623, 1976. [p. 10]

[30] A.M. Danilevski. Sur la solution numérique de l'équation caractéristique (en russe). *Matem. Sbornik*, 44(2):169–172, 1937. [p. 9]

[31] J.D. Dixon. Exact solution of linear equations using p-adic expansions. *Numer. Math.*, 40:137–141, 1982. [p. 11]

[32] J.L. Dornstetter. On the equivalence between Berlekamp's and Euclid's algorithms. *IEEE Trans. Inform. Theory*, IT-33:428–431, 1987. [p. 18]

[33] J.G. Dumas. Algorithmes parallèles efficaces pour le calcul formel: algèbre linéaire creuse et extensions algébriques. Thèse de Doctorat, Institut National Polytechnique de Grenoble, France, décembre 2000. [pp. 21, 42, 43]

[34] J.G. Dumas, T. Gautier, M. Giesbrecht, P. Giorgi, B. Hovinen, E. Kaltofen, B.D. Saunders, W.J. Turner, and G. Villard. LinBox: A Generic Library for Exact Linear Algebra. In *International Congress of Mathematical Software (ICMS 2002), World Scientific, Beijing, China, August 17-19*, pages 40–50, 2002. [pp. 21, 42]

[35] J.G. Dumas, B.D. Saunders, and G. Villard. On efficient sparse integer matrix Smith normal form computations. *Journal of Symbolic Computation,* special issue on *Computer Algebra and Mechanized Reasoning*, 32(1-2):71–99, 2001. [p. 48]

[36] W. Eberly. Very fast parallel matrix and polynomial arithmetic. MSc Thesis, Department of Computer Science, University of Toronto, Canada. TR 178/85, 1985. [p. 10]

[37] W. Eberly. Asymptotically Efficient Algorithms for the Frobenius Form. Technical report, Department of Computer Science, University of Calgary, Canada, TR2000/649/01, Jan. 2000. [p. 46]

[38] W. Eberly. Black box Frobenius decompositions over small fields. In *International Symposium on Symbolic and Algebraic Computation, St Andrews, Scotland*, pages 106–113, Aug 2000. [p. 46]

[39] W. Eberly. Private communication, 2000. [p. 21]

[40] W. Eberly, M. Giesbrecht, and G. Villard. Computing the determinant and Smith form of an integer matrix. In *The 41st Annual IEEE Symposium on Foundations of Computer Science, Redondo Beach, CA*, pages 675–685. IEEE Computer Society Press, November 2000. [pp. 3, 27, 28]

[41] W. Eberly and E. Kaltofen. On randomized Lanczos algorithms. In *International Symposium on Symbolic and Algebraic Computation, Maui, Hawaii, USA*, pages 176–183. ACM Press, July 1997. [pp. 20, 21, 42, 44, 45]

[42] J. Edmonds. Systems of distinct representatives and linear algebra. *J. Res. Nat. Bur. Standards – B*, 71(4):241–245, 1967. [p. 9]

[43] D.K. Faddeev and I.S. Sominskii. *Problèmes de l'algèbre supérieure (en russe)*. 2^{e} éd. Moscou, 1949. [p. 10]

[44] V.N. Faddeeva. *Computational methods of linear algebra*. Dover Publications, Inc. New-york, 1959. [p. 10]

[45] G.D. Forney. Minimal bases of rational vector spaces, with applications to multivariable linear systems. *SIAM J. Control*, 13:493–520, 1975. [p. 32]

[46] J.S. Frame. A simple recurrent formula for inverting a matrix. *Bull. American Math. Soc.*, 55:1045, 1949. [p. 10]

[47] G. Frobenius. Über der Rang einer Matrix. *Sitzungsberichte der Königlich Preussischen Akademie der Wissenschaften zu Berlin*, pages 54–65, 1911. [p. 8]

[48] Z. Galil and V.Y. Pan. Parallel evaluation of the determinant and of the inverse of a matrix. *Inform. Proc. Lett.*, 30:41–45, 1989. [p. 10]

[49] F.R. Gantmacher. *Théorie des matrices*. Dunod, Paris, France, 1966. [pp. iv, 2, 4, 8, 9, 10]

[50] J. von zur Gathen. Hensel and Newton methods in valuation rings. *Math. Comp.*, 42(166):637–661, 1984. [p. 3]

[51] J. von zur Gathen and J. Gerhard. *Modern Computer Algebra*. Cambridge University Press, 1999. [pp. iv, 2, 3, 6, 16, 20, 30]

[52] M. Giesbrecht. *Nearly optimal algorithms for canonical matrix forms*. PhD thesis, Department of Computer Science, University of Toronto, 1993. [pp. 9, 10]

[53] M. Giesbrecht. Nearly optimal algorithms for canonical matrix forms. *SIAM Journal on Computing*, 24(5):948–969, 1995. [p. 9]

[54] M. Giesbrecht. Efficient parallel solution of sparse systems of linear diophantine equations. In *Second International Symposium on Parallel Symbolic Computation (PASCO'97), Maui, Hawaii, USA*, pages 1–10, Jul 1997. [p. 48]

[55] M. Giesbrecht. Fast computation of the Smith form of a sparse integer matrix. *Computational Complexity*, 10:41–69, 2001. [pp. 30, 48]

[56] M. Giesbrecht, A. Lobo, and B.D. Saunders. Certifying inconsistency of sparse linear systems. In *International Symposium on Symbolic and Algebraic Computation, Rostock, Germany*, pages 113–119. ACM Press, August 1998. [p. 45]

[57] M. Giesbrecht and A. Storjohann. Computing rational forms of integer matrices. *Journal of Symbolic Computation*, 34(3):157–172, 2002. [pp. 10, 30]

[58] I. Gohberg, T. Kailath, and I. Koltracht. Efficient solution of linear systems of equations with recursive structure. *Linear Algebra and its Applications*, 80:81–113, 1986. [p. 6]

[59] I. Gohberg, P. Lancaster, and L. Rodman. *Matrix polynomials*. Academic Press, New-York, 1982. [p. v]

[60] C. Hermite. Sur l'introduction des variables continues dans la théorie des nombres. *J. Reine Angew. Math.*, 41:191–216, 1851. [p. 9]

[61] X. Huang and V.Y. Pan. Fast rectangular matrix multiplications and improving parallel matrix computations. In *Second International Symposium on Parallel Symbolic Computation (PASCO'97), Maui, Hawaii, USA*, pages 11–23, Jul 1997. [p. 25]

[62] X. Huang and V.Y. Pan. Fast rectangular matrix multiplication and applications. *J. of Complexity*, 14:257–299, 1998. [p. 1]

[63] C.P. Jeannerod and G. Villard. Straight-line computation of the polynomial matrix inverse. Research Report 2002-47, Laboratoire LIP, ENS Lyon, France, 2002. [pp. 35, 37]

[64] T. Kailath. *Linear systems*. Prentice Hall, 1980. [pp. v, 3, 4, 7, 8, 51, 52]

[65] R.E. Kalman. Irreducible realizations and the degree of a rational matrix. *SIAM J. Appl. Math.*, 13:520–544, 1965. [p. 4]

[66] E. Kaltofen. On computing determinants without divisions. In *International Symposium on Symbolic and Algebraic Computation, Berkeley, California USA*, pages 342–349. ACM Press, July 1992. [pp. 18, 23, 24, 28]

[67] E. Kaltofen. Analysis of Coppersmith's block Wiedemann algorithm for the parallel solution of sparse linear systems. *Math. Comp.*, 64(210):777–806, 1995. [pp. 6, 13, 16, 17, 20, 42, 45]

[68] E. Kaltofen. Challenges of symbolic computation: my favorite open problems. *J. Symbolic Computation*, 29(6):891–919, 2000. [pp. 41, 46]

[69] E. Kaltofen. Private communication, 2000. [pp. 27, 28]

[70] E. Kaltofen. An output-sensitive variant of the baby steps/giant steps determinant algorithm. In *International Symposium on Symbolic and Algebraic Computation, Lille, France*. ACM Press, July 2002. [p. 29]

[71] E. Kaltofen. Efficient problem reductions in linear algebra. In *The 8th International IMACS Conference on Applications of Computer Algebra, Volos, Greece*, June 2002. [p. 45]

[72] E. Kaltofen and V.Y. Pan. Processor efficient parallel solution of linear systems over an abstract field. In *Proc. 3rd Annual ACM Symposium on Parallel Algorithms and Architecture*, pages 180–191. ACM-Press, 1991. [p. 10]

[73] E. Kaltofen and V.Y. Pan. Processor efficient parallel solution of linear systems II: the general case. In *Proc. 33rd IEEE Symp. Foundations Of Computer Science, Pittsburg, USA*, pages 714–723, 1992. [p. 10]

[74] E. Kaltofen and B.D. Saunders. On Wiedemann's method of solving sparse linear systems. In *Proc. AAECC-9*, LNCS 539, Springer Verlag, pages 29–38, 1991. [pp. 10, 15, 42, 44, 45]

[75] E. Kaltofen and B. Trager. Greatest common divisors, factorization, separation of numerators and denominators. *J. Symbolic Computation*, 9(3):301–320, 1990. [p. 41]

[76] E. Kaltofen and G. Villard. Computing the sign or the value of the determinant of an integer matrix, a complexity survey. *J. Comp. Applied Math.* to appear. [pp. 27, 28]

[77] E. Kaltofen and G. Villard. On the complexity of computing determinants. In K. Shirayanagi and K. Yokoyama, editors, *Proc. Fifth Asian Symposium on Computer Mathematics (ASCM 2001)*, volume 9 of *Lecture Notes Series on Computing*, pages 13–27, Singapore, 2001. World Scientific. Maple 6 complexity worksheet `http://www.math.ncsu.edu/~kaltofen/`. [pp. 13, 14, 17, 24, 26, 29]

[78] W. Keller-Gehrig. Fast algorithms for the characteristic polynomial. *Theoretical Computer Science*, 36:309–317, 1985. [pp. 1, 9]

[79] D.E. Knuth. The analysis of algorithms. In *Proc. International Congress of Mathematicians, Nice, France*, volume 3, pages 269–274, 1970. [pp. 6, 20]

[80] P. Koiran, N. Portier, and G. Villard. A rank theorem for Vandermonde matrices. Research Report 2001-34, Laboratoire LIP, ENS Lyon, France, 2001. [p. 6]

[81] A.N. Krylov. Sur la résolution numérique de l'équation déterminant la fréquence des petites oscillations dans les problèmes techniques (en russe). *Izv. Akad. Nauk SSSR, Ser. Fiz. Mat.*, 4:491–539, 1931. [p. 10]

[82] S.E. Labhalla, H. Lombardi, and R. Marlin. Algorithmes de calcul de la réduction d'Hermite d'une matrice à coefficients polynomiaux. *Theoretical Computer Science*, 161(1–2):69–92, 1995. [p. 31]

[83] R. Lambert. *Computational aspects of discrete logarithms.* PhD thesis, University of Waterloo, Ontario, Canada, 1996. [pp. 21, 42]

[84] C. Lanczos. Solutions of systems of linear equations by minimized iterations. *J. Res. Bur. Standards, Sect. B*, 49:33–53, 1952. [p. 10]

[85] U. Le Verrier. Sur les variations séculaires des éléments elliptiques des sept planètes principales: Mercure, Vénus, La Terre, Mars, Jupiter, Saturne et Uranus. *J. Math. Pures Appl.*, 4:220–254, 1840. [p. 10]

[86] C.C. MacDuffee. *The theory of matrices.* Chelsea, New-York, 1956. [pp. iv, 9]

[87] M. Mahajan and V. Vinay. Determinant: old algorithms, new insights. *SIAM J. Discrete Math.*, 12(4):474–490, 1999. [p. 10]

[88] M. Mahajan and V. Vinay. Determinant: Combinatorics, Algorithms, and Complexity. *Chicago Journal of Theoretical Computer Science*, Article 5, 1997. [p. 10]

[89] J.L. Massey. Shift-register synthesis and BCH decoding. *IEEE Trans. Inform. Theory*, 15:122–127, 1969. [p. 6]

[90] M.H. Mathieu and D. Ford. On p-adic computation of the rational form of a matrix. *J. Symbolic Computation*, 10:453–464, 1990. [p. 30]

[91] R.T. Moenck and J.H. Carter. Approximate algorithms to derive exact solutions to systems of linear equations. In *Proc. EUROSAM*, LNCS 72, Springer Verlag, pages 63–73, 1979. [p. 11]

[92] M. Morf. Doubling algorithms for Toeplitz and related equations. In *IEEE Internat. Conf. Acoust. Speech Signal Process., Piscataway, NJ*, pages 954–959, 1980. [p. 6]

[93] T. Mulders and A. Storjohann. Certified dense linear system solving. *Journal of Symbolic Computation*. To appear. [pp. 11, 48]

[94] T. Mulders and A. Storjohann. On lattice reduction for polynomial matrices. *Journal of Symbolic Computation*. To appear. [pp. 32, 33]

[95] K. Mulmuley. A fast parallel algorithm to compute the rank of a matrix over an arbitrary field. *Combinatorica*, 7(1):101–104, 1987. [p. 44]

[96] V.C. Nanda. Arithmetic functions of matrices and polynomial identities. In *Colloq. Math. Soc. János Bolyai 34, Budapest, Hungary*, pages 1107–1126. North-Holland, 1981. [p. 15]

[97] M. Newman. *Integral Matrices*. Academic Press, 1972. [pp. iv, 2, 3, 9]

[98] P Ozello. *Calcul exact des formes de Jordan et de Frobenius d'une matrice*. Thèse de Doctorat, Université Scientifique et Médicale de Grenoble, France, 1987. [p. 9]

[99] V.Y. Pan. Complexity of parallel matrix computations. *Theoretical Computer Science*, 54:65–85, 1987. [p. 27]

[100] V.Y. Pan. Computing the determinant and the characteristic polynomial of a matrix via solving linear systems of equations. *Inf. Proc. Letters*, 28:71–75, 1988. [p. 27]

[101] V.Y. Pan and Z.Q. Chen. The complexity of the matrix eigenproblem. In *The 21rst Annual ACM Symposium on Theory of Computing, Atlanta, Georgia*, pages 507–516. ACM Press, May 1999. [p. 46]

[102] V.Y. Pan, Z.Q. Chen, and A. Zheng. The complexity of the algebraic eigenproblem. MSRI Preprint 1998-071, Mathematical Sciences Research Institute, California, USA, dec. 1998. [p. 46]

[103] V.M. Popov. Some properties of control systems with irreducible matrix transfer functions. In *Lecture Notes in Mathematics*, pages 169–180. Vol. 144, Springer, Berlin, 1969. [p. 3]

[104] F.P. Preparata and D.V. Sarwate. An improved parallel processor bound in fast matrix inversion. *Inform. Proc. Lett.*, 7:148–150, 1978. [p. 10]

[105] M.P. Quéré-Stuchlik. *Algorithmique des faisceaux linéaires de matrices, application à la théorie des systèmes linéaires et à la résolution d'équations algébro-différentielles*. Thèse de Doctorat, Université de Paris VI, France, 1997. [p. 33]

[106] J.H. Reif. Efficient parallel computation of the characteristic polynomial of a sparse, separable matrix. *Algorithmica*, 29(3):487–510, 2001. [p. 46]

[107] P. Samuelson. A method of determining explicitly the coefficients of the characteristic equation. *Ann. Math. Stat.*, 13:424–429, 1942. [p. 10]

[108] B.D Saunders, A. Storjohann, and G. Villard. Matrix rank certification. *Elect. J. Linear Algebra*. To appear. [p. 44]

[109] A. Schönhage. Schnelle Berechnung von Kettenbruchenwicklungen. *Acta Informatica*, 1:139–144, 1971. [pp. 6, 20]

[110] A. Schönhage. Unitäre Transformation grosser Matrizen. *Num. Math.*, 20:409–417, 1973. [p. 1]

[111] H.J.S. Smith. On systems of linear indeterminate equations and congruences. *Philos. Trans. Royal Soc. London*, 151:293–326, 1861. [pp. 2, 9]

[112] M. Soltys. Berkowitz's algorithm and clow sequences. *The Electronic Journal of Linear Algebra*, 9:42–54, 2002. [p. 10]

[113] J.M. Souriau. Une méthode pour la décomposition spectrale et l'inversion des matrices. *Comptes Rendus de l'Académie des Sciences*, 227:1010–1011, 1948. [p. 10]

[114] A. Storjohann. *Algorithms for Matrix Canonical Forms.* PhD thesis, Institut für Wissenschaftliches Rechnen, ETH-Zentrum, Zurich, Switzerland, November 2000. [pp. 8, 9, 32, 33]

[115] A. Storjohann. Computing the Frobenius form of a sparse integer matrix, manuscript, April 2000. [pp. 30, 48]

[116] A. Storjohann. High-Order Lifting (Extended Abstract). In *International Symposium on Symbolic and Algebraic Computation, Lille, France*, pages 246–254. ACM Press, July 2002. [pp. 11, 17, 27, 33, 34, 51]

[117] A. Storjohann and G. Villard. Algorithms for similarity transforms. In *The Seventh Rhine Workshop on Computer Algebra, Bregenz, Austria*, March 2000. [p. 8]

[118] V. Strassen. Gaussian elimination is not optimal. *Numer. Math.*, 13:354–356, 1969. [p. 1]

[119] V. Strassen. Vermeidung von Divisionen. *J. Reine Angew. Math.*, 264 :182–202, 1973. [pp. 1, 23]

[120] E. Thomé. Fast computation of linear generators for matrix sequences and application to the block Wiedemann algorithm. In *International Symposium on Symbolic and Algebraic Computation, London, Ontario*, pages 323–331. ACM Press, July 2001. [p. 17]

[121] W.J. Turner. *Black box linear algebra with the LinBox library.* PhD thesis, North Carolina State University, Raleigh, NC USA, May 2002. [pp. 6, 44]

[122] L.G. Valiant. Why is Boolean complexity theory difficult ? In M.S. Paterson, editor, *Boolean Function Complexity*, volume 169 of *London Math. Soc. Lecture Notes Series*, pages 84–94, Cambridge, 1992. Cambridge University Press. [p. 10]

[123] G. Villard. Generalized subresultants for computing the Smith normal form of polynomial matrices. *Journal of Symbolic Computation*, 20:269–286, 1995. [p. 31]

[124] G. Villard. Computing Popov and Hermite forms of polynomial matrices. In *International Symposium on Symbolic and Algebraic Computation, Zurich, Suisse*, pages 250–258. ACM Press, July 1996. [pp. 3, 6, 32, 33, 34]

[125] G. Villard. A study of Coppersmith's block Wiedemann algorithm using matrix polynomials, Feb. 1997. RR 975-I-M IMAG Grenoble, France. [pp. 13, 15, 16, 55]

[126] G. Villard. Fast parallel algorithms for matrix reduction to normal forms. *Appli. Alg. Eng., Comm. Comp.*, 8(6):511–537, 1997. [pp. 8, 10]

[127] G. Villard. Further analysis of Coppersmith's block Wiedemann algorithm for the solution of sparse linear systems. In *International Symposium on Symbolic and Algebraic Computation, Maui, Hawaii, USA*, pages 32–39. ACM Press, July 1997. [pp. 5, 13, 15, 16, 17, 20]

[128] G. Villard. Block solution of sparse linear systems over GF(q): the singular case. SIGSAM *Bulletin*, 32(4):10–12, 1999. [p. 45]

[129] G. Villard. Block algorithms for the characteristic polynomial, manuscript, LinBox Group, Feb. 2000. [p. 46]

[130] G. Villard. Computing the Frobenius form of a sparse matrix. In *The Third International Workshop on Computer Algebra in Scientific Computing, Samarkand, Uzbekistan*, pages 395–407. Springer-Verlag, October 2000. [pp. 27, 28, 46, 47]

[131] G. Villard. Processor efficient parallel solution of linear systems of equations. *Journal of Algorithms*, 35(1):122–126, 2000. [p. 10]

[132] G. Villard. Column reduction of matrix polynomials, manuscript, October 2002. [p. 35]

[133] G. Villard. Exact computation of the determinant and of the inverse of a matrix, August 2002. *Workshop on Complexity, Foundations of Computational Mathematics FoCM'02, Minneapolis.* [p. 35]

[134] D. Wiedemann. Solving sparse linear equations over finite fields. *IEEE Transf. Inform. Theory*, IT-32:54–62, 1986. [pp. 10, 13, 15, 17, 42, 43, 44, 45]

Index