Un algorithme pour le calcul des syzygies sur $\mathbf{V}[X]$ dans le cas où $\mathbf{V}$ est un domaine de valuation

Henri Lombardi (*), Claude Quitté (†), Ihsen Yengui (‡)

Février 2012

Résumé

Soit $\mathbf{V}$ un domaine de valuation. Nous donnons un algorithme pour calculer une base du $\mathbf{V}$-saturé d'un sous-module de type fini d'un $\mathbf{V}$-module libre (avec une base éventuellement infinie). Nous l'appliquons pour calculer le $\mathbf{V}$-saturé d'un sous-$\mathbf{V}[X]$-module de type fini de $\mathbf{V}[X]^n$ ($n \in \mathbb{N}^*$). Ceci permet enfin de calculer un système générateur fini pour les syzygies sur $\mathbf{V}[X]$ d'une famille finie de vecteurs de $\mathbf{V}[X]^k$.

Mots clés : Saturation, Cohérence, Syzygies, Anneau de valuation, Calcul formel, Algèbre constructive.

Abstract

We give an algorithm for computing the $\mathbf{V}$-saturation of any finitely generated submodule of $\mathbf{V}[X]^n$ ($n \in \mathbb{N}^*$), where $\mathbf{V}$ is a valuation domain. This allows us to compute a finite system of generators for the syzygy module of any finitely generated submodule of $\mathbf{V}[X]^k$.

Key words: Saturation, Coherence, Syzygies, Valuation domains, Computer algebra, Constructive Algebra.

Introduction

Cela fait partie du folklore (voir par exemple [3, Glaz, Th. 7.3.3]) que pour un domaine de valuation $\mathbf{V}$, l'anneau $\mathbf{V}[X]$ est cohérent, (i.e., le module des syzygies pour un idéal de type fini de $\mathbf{V}[X]$ est de type fini). Néanmoins la preuve dans la référence citée découle d'un résultat profond et difficile dans un gros article de Gruson et Raynaud [4]. Et il semble en outre qu'il n'existe pas d'algorithme dans la littérature existante pour ce résultat remarquable.

Dans le cas des anneaux nœthériens cohérents (non nécessairement des domaines de valuation), on sait que l'anneau de polynômes $\mathbf{R}[X_1, \ldots, X_n]$ est également nœthérien cohérent. Une preuve constructive se trouve dans [9] et est exposée dans le livre [8]. On peut aussi dans ce cas utiliser les bases de Gröbner, introduites par Buchberger pour les anneaux de polynômes sur des corps (voir par exemple [1, 5, 10]).

Néanmoins, la nœthérianité n'est pas le fin mot de l'affaire puisque le résultat concernant la cohérence dans le cas des corps s'étend facilement aux anneaux zéro-dimensionnels réduits (aussi appelés Von Neuman réguliers, ou absolument plats).

Dans [7], un algorithme est donné pour le calcul d'une base de Gröbner pour un idéal de type fini de $\mathbf{V}[X]$ dans le cas d'un domaine de valuation de dimension 1 (non nécessairement nœthérien). On peut en déduire un algorithme pour le calcul d'un système générateur fini pour les syzygies d'un idéal de type fini de $\mathbf{V}[X]$.

*Équipe de Mathématiques, UMR CNRS 6623, UFR des Sciences et Techniques, Université de Franche-Comté, 25030 Besançon cedex, FRANCE, email: henri.lombardi@univ-fcomte.fr

†Laboratoire de Mathématiques, SP2MI, Boulevard 3, Téléport 2, BP 79, 86960 Futuroscope Cedex, France, email: claude.quitte@math.univ-poitiers.fr

‡Departement of Mathematics, Faculty of Sciences of Sfax, 3000 Sfax, Tunisia, email: ihsen.yengui@fss.rnu.tn

Rappelons (voir par exemple [8]) que sur un anneau cohérent tout module de présentation finie M est un module cohérent (i.e., le module des syzygies pour un sous-module de type fini de M est de type fini).

Dans l'article présent, nous donnons un algorithme débarrassé de toute hypothèse nœthérienne, ainsi que de toute hypothèse concernant la dimension de Krull, pour le calcul d'un système générateur fini des syzygies d'une famille finie de vecteurs de $\mathbf{V}[X]^k$. Nous pensons fournir ainsi par la même occasion la première preuve constructive du résultat (car notre algorithme est prouvé constructivement).

Rappelons d'abord que pour un sous-$\mathbf{R}$-module M d'un module N avec $\mathbf{R}$ un anneau intègre, le $\mathbf{R}$-*saturé de* M *dans* N est le $\mathbf{R}$-module

$$\mathrm{Sat}_{\mathbf{R},N}(M) = \{\, x \in N \mid \exists a \in \mathbf{R}^*,\, ax \in M \,\}.$$

S'il y a plusieurs anneaux en présence, on dira pour préciser le "$\mathbf{R}$-saturé de M dans N". Dans le cas où N est un module libre (de la forme $\mathbf{R}^n$, $n \in \mathbb{N}$ ou $\mathbf{R}^{(I)}$, I infini) on obtient par extension des scalaires un $\mathbf{K}$-espace vectoriel $\mathbf{K} \otimes N \simeq \mathbf{K}^n$ ou $\mathbf{K}^{(I)}$, où $\mathbf{K}$ est le corps des fractions de $\mathbf{R}$. On a alors aussi $\mathrm{Sat}_{\mathbf{R},N}(M) = \mathbf{K}.M \cap N$, où $\mathbf{K}.M$ est le sous-$\mathbf{K}$-espace vectoriel de $\mathbf{K} \otimes N$ engendré par M. Le module M est dit $\mathbf{R}$-saturé s'il est égal à son $\mathbf{R}$-saturé. En général, le $\mathbf{R}$-saturé d'un $\mathbf{R}[X]$-module de type fini dans $N = \mathbf{R}[X]^n$ n'a aucune raison d'être lui-même un $\mathbf{R}[X]$-module de type fini, mais c'est ce qui arrive si l'anneau $\mathbf{R}$ est un domaine de valuation $\mathbf{V}$.

Nous donnons dans la section 1 un algorithme incrémental pour calculer une base du $\mathbf{V}$-saturé d'un sous-module de type fini d'un $\mathbf{V}$-module libre (avec une base éventuellement infinie). Cet algorithme n'est pas un scoop, mais il est mis dans une forme où nous sommes capables de l'utiliser, dans la section 2, pour calculer un système générateur fini du $\mathbf{V}[X]$-module obtenu en $\mathbf{V}$-saturant un sous-$\mathbf{V}[X]$-module de type fini de $\mathbf{V}[X]^n$ ($n \in \mathbb{N}^*$).

Ceci démontre que le $\mathbf{V}$-saturé d'un $\mathbf{V}[X]$-module de type fini dans $\mathbf{V}[X]^n$ est bien un $\mathbf{V}[X]$-module de type fini.

Enfin on obtient comme conséquence immédiate, dans la section 3, le calcul d'un système générateur fini pour les syzygies sur $\mathbf{V}[X]$ d'une famille finie de vecteurs de $\mathbf{V}[X]^k$.

Dans cet article tous les anneaux sont commutatifs et unitaires.

1 Saturation d'un V-module de type fini dans un V-module libre

Terminologie : Nous utiliserons dans cet article la terminologie usuelle de l'algèbre constructive comme dans [8], bien adaptée au Calcul Formel.

Pour un anneau $\mathbf{R}$ arbitraire on note $\mathbf{R}^\times$ le groupe multiplicatif des unités de $\mathbf{R}$. L'anneau $\mathbf{R}$ est dit *discret* lorsque l'on a un algorithme qui décide si $x = 0$ ou $x \neq 0$ pour un élément arbitraire de $\mathbf{R}$. Un anneau $\mathbf{R}$ est dit *local* lorsque l'on a de façon explicite l'implication

$$\forall x, y \in \mathbf{R},\ x + y \in \mathbf{R}^\times \implies (x \in \mathbf{R}^\times \ \vee\ y \in \mathbf{R}^\times)$$

Il revient au même de demander

$$\forall x \in \mathbf{R}, (x \in \mathbf{R}^\times \ \vee\ 1 + x \in \mathbf{R}^\times)$$

Un anneau local non nul $\mathbf{R}$ admet pour unique idéal maximal son *radical de Jacobson*

$$\mathrm{Rad}(\mathbf{R}) = \left\{\, x \in \mathbf{R} \mid 1 + x\mathbf{R} \subseteq \mathbf{R}^\times \,\right\}$$

(pour un anneau arbitraire $\mathbf{R}$ cet idéal est égal à l'intersection des idéaux maximaux lorsqu'on se situe en mathématiques classiques).

L'anneau quotient $\mathbf{k} = \mathbf{R}/\mathrm{Rad}(\mathbf{R})$ est un corps, appelé *corps résiduel* de $\mathbf{R}$. L'anneau local $\mathbf{R}$ est dit *résiduellement discret* lorsque l'on a de façon explicite la disjonction $\forall x \in \mathbf{R}$, $(x \in \mathbf{R}^\times \vee x \in \mathrm{Rad}(\mathbf{R}))$. Dans ce cas le corps résiduel est un corps discret. On a alors un algorithme qui décide la disjonction "$x = 0$ ou x inversible" pour tout $x \in \mathbf{k}$.

Soit I un ensemble, fini ou infini, muni d'une relation d'ordre total "discrète", i.e. nous disposons d'un algorithme qui décide la disjonction

$$i < j \quad \vee \quad i = j \quad \vee \quad i > j$$

pour $i, j \in I$. On peut alors, pour n'importe quel anneau $\mathbf{R}$, considérer le module libre $\mathbf{R}^{(I)}$ dans lequel on notera $(\mathrm{e}_i)_{i\in I}$ la base naturelle. Si $\mathbf{R}$ est un anneau discret tout vecteur $a = \sum_{i\in J} a_i \mathrm{e}_i$ dans $\mathbf{R}^{(I)}$ (J est une partie finie de I) peut être testé nul ou non nul, et lorsqu'il est non nul, on peut déterminer le plus petit indice i pour lequel $a_i \neq 0$.

Pour déterminer un sous-$\mathbf{R}$-module saturé d'un module libre $\mathbf{R}^{(I)}$ on fera appel aux lemmes 1 et 3 suivants.

Lemme 1 *Soit $\mathbf{R}$ un anneau intègre et N un $\mathbf{R}$-module sans torsion. Si $N = M \oplus P$ alors M est saturé dans N.*

Dans toute la suite de cette section, on suppose que $\mathbf{R}$ est un anneau local intègre résiduellement discret avec $\mathbf{k} = \mathbf{R}/\mathrm{Rad}(\mathbf{R})$.

Définition et notation 2

1. Un vecteur $C = \sum_i c_i \mathrm{e}_i$ de $\mathbf{R}^{(I)}$ (que nous voyons comme un vecteur colonne) est dit *primitif* s'il est résiduellement non nul, i.e., l'un de ses coefficients est une unité. Dans ce cas-là nous notons $\mathrm{piv}(C)$ le plus petit indice i pour lequel c_i est résiduellement non nul. C'est *l'indice pivot de C*. Nous notons $\mathrm{cpiv}(C)$ (*coefficient pivot de C*) le scalaire c_i correspondant.
2. Une famille finie $(C^k)_{k\in K}$ de vecteurs primitifs est dite $\mathbf{k}$-*échelonnée* si les $\mathrm{piv}(C^k)$ sont deux à deux distincts. Une famille $\mathbf{k}$-échelonnée est aussi appelée simplement *échelonnée*. On note alors
$$\mathrm{piv}(C) = \left\{ \mathrm{piv}(C^k) \mid k \in K \right\}.$$
Une matrice à coefficients dans $\mathbf{R}$ sera dite échelonnée si ses vecteurs colonnes sont primitifs et forment une famille échelonnée.
3. La famille $(C^k)_{k\in K}$ est dite *en forme* $\mathbf{k}$-*échelonnée stricte*, lorsque K est un ensemble totalement ordonné, si elle est échelonnée et si en outre, pour $j < k$ dans K le coefficient d'indice $\mathrm{piv}(C^j)$ de C^k est nul.

Lemme 3 *Si une famille finie $C = (C^k)_{k\in K}$ de vecteurs de $\mathbf{R}^{(I)}$ est $\mathbf{k}$-échelonnée, elle forme une base du $\mathbf{R}$-module M qu'elle engendre et M admet pour supplémentaire le $\mathbf{R}$-module libre*

$$P = \bigoplus\nolimits_{j\in J} \mathbf{R}\,\mathrm{e}_j \quad \textit{avec } J = \{\, j \in I \mid j \notin \mathrm{piv}(C) \,\}.$$

En outre l'ensemble $\mathrm{piv}(C)$ *ne dépend que du module M. En effet un indice j est dans* $\mathrm{piv}(C)$ *si et seulement si il existe un vecteur primitif U dans M avec* $\mathrm{piv}(U) = j$.

NB : les deux $\mathbf{R}$-modules libres en question sont automatiquement $\mathbf{R}$-saturés d'après le lemme 1.

Démonstration. Nous donnons l'argument lorsque I est fini, mais il s'adapte facilement au cas général. Si on ordonne la famille formée des e_j pour $j \in J$ et des C^k pour $k \in K$, en ordre d'indices pivots croissants. Alors la matrice formée par ces colonnes est résiduellement triangulaire avec des coefficients inversibles sur la diagonale, donc elle est résiduellement inversible, donc elle est inversible. Ceci montre que M et P sont supplémentaires et admettent les bases voulues. Le reste est laissé au lecteur. □

L'algorithme de saturation

Contexte 4 *Soit* $\mathbf{V}$ *un domaine de valuation, i.e. un anneau intègre dans lequel pour tous* a, b *on a* $a \mid b$ *ou* $b \mid a$*, i.e. plus précisément on a un algorithme qui décide (pour* a, b *donnés dans* $\mathbf{V}$*) la disjonction*

$$\exists x \in \mathbf{V},\ a = xb \quad \vee \quad \exists x \in \mathbf{V},\ b = xa$$

et fournit l'élément x*. On sait que* $\mathbf{V}$ *est un anneau local (supposons* $a + b$ *inversible, si* a *divise* b *il divise* $a + b$ *et il est inversible, si* b *divise* a *il divise* $a + b$ *et il est inversible). On note* $\mathbf{K}$ *son corps de fractions et* $\mathbf{k}$ *son corps résiduel. Comme* $\mathbf{V}$ *est supposé intègre de manière explicite, le corps* $\mathbf{K}$ *est un corps discret. On suppose en outre que* $\mathbf{V}$ *est* résiduellement discret*, ce qui signifie que l'on a un algorithme pour décider si un élément de* $\mathbf{V}$ *est une unité. En particulier les lemmes 1 et 3 sont satisfaits avec l'anneau* $\mathbf{V}$*.*

Remarque. Dans un domaine de valuation résiduellement discret, on a un test pour répondre à la question « $a \mid b$? ». En effet, pour a, b non nuls, si $a = bx$, alors $a \mid b$ si et seulement si x est une unité.

On considère un sous-$\mathbf{V}$-module $M = \mathbf{V}\,a^1 + \cdots + \mathbf{V}\,a^m$ de $\mathbf{V}^{(I)}$. L'objectif de cette section est de donner un algorithme pour calculer une base du $\mathbf{V}$-saturé de M dans $\mathbf{V}^{(I)}$, $\mathbf{V}$-module que nous notons de manière abrégée $\mathrm{Sat}(M)$. En fait, comme seul un nombre fini d'indices sont en cause, on peut aussi bien supposer que I est fini et que M est engendré par les colonnes d'une matrice F. Pour visualiser la chose nous pouvons écrire les lignes de la matrice en ordre décroissant pour les indices.

Une manière brutale de calculer $\mathrm{Sat}(M)$ serait de réduire F à la forme de Smith par des manipulations élémentaires. On voit alors qu'après un changement de base convenable, le module M est engendré par des vecteurs $v_i f_i$ (où les f_i forment une partie d'une base et les v_i sont non nuls). Dans ces conditions, la module $\mathrm{Sat}(M)$ est simplement le module engendré par ces f_i. Ceci nous indique que $\mathrm{Sat}(M)$ est un $\mathbf{V}$-module libre ayant pour supplémentaire un autre $\mathbf{V}$-module libre.

En fait nous préférons procéder de manière moins brutale et obtenir une base de $\mathrm{Sat}(M)$ comme les vecteurs colonnes d'une matrice $\mathbf{k}$-échelonnée G que nous calculons à partir de F au moyen d'opérations très simples.

Une première opération, de réduction d'un vecteur, que nous noterons $\mathsf{RedPrim}$ consiste à remplacer un vecteur a, supposé non nul, par a/u, où u est un pgcd de ses coefficients, par exemple u est le coefficient d'indice minimum parmi ceux qui divisent tous les autres, auquel cas le coefficient pivot du vecteur réduit est égal à 1.

Le traitement que nous faisons subir à la matrice F pour la ramener à une forme $\mathbf{k}$-échelonnée stricte G telle que $\mathrm{Sat}(\mathrm{Im}(F)) = \mathrm{Im}(G)$ est la suivante. Elle procède en traitant une après l'autre les colonnes de la matrice initiale. Notez qu'au départ, la matrice vide est en forme $\mathbf{k}$-échelonnée stricte.

Supposons qu'on ait traité quelques colonnes initiales de la matrice et qu'on ait obtenu une matrice $\mathbf{k}$-échelonnée stricte avec pour colonnes $C^1, \ldots, C^r$.

On veut traiter une nouvelle colonne, que l'on appelle $C = \sum_i c_i \mathrm{e}_i$. On procède comme suit.

1. Pivot de Gauss : on opère des manipulations élémentaires de colonnes classiques

$$C \leftarrow C - \frac{c_s}{c_{j,s}}\, C^j,$$

   ici $s = \mathrm{piv}(C^j)$ et $c_{j,s} = \mathrm{cpiv}(C^j)$. Cette opération est faite successivement avec les colonnes $C^1, \ldots, C^r$. On obtient alors une colonne C'.

2. Si $C' = 0$, on ne la rajoute pas. La matrice reste en forme $\mathbf{k}$-échelonnée stricte. Le $\mathbf{K}$-espace vectoriel engendré par les colonnes $C^1, \ldots, C^r, C$ admet $(C^1, \ldots, C^r)$ pour base.

3. Si $C' \neq 0$, on remplace C' par sa forme réduite primitive $C'' = \mathsf{RedPrim}(C')$. Nous rajoutons alors C'' comme dernière colonne C^{r+1} de la matrice. Et la nouvelle matrice est en forme $\mathbf{k}$-échelonnée stricte. Le $\mathbf{K}$-espace vectoriel engendré par les colonnes $C^1, \ldots, C^r, C$ admet $(C^1, \ldots, C^r, C'')$ pour base.

Par construction $\mathrm{Im}(G)$ est contenu dans $\mathrm{Sat}(\mathrm{Im}(F))$ et le $\mathbf{V}$-module $\mathrm{Im}(G)$ est saturé parce que G est en forme $\mathbf{k}$-échelonnée stricte. En fait, aussi bien dans le cas 2. que dans le cas 3., on voit par récurrence que l'on a bien construit une base du $\mathbf{V}$-saturé engendré par les première colonnes (jusqu'à la colonne C). À la fin du processus on a donc $\mathrm{Sat}(\mathrm{Im}(G)) = \mathrm{Sat}(\mathrm{Im}(F))$. Notre algorithme remplit bien le but fixé.

Théorème 5
L'algorithme de saturation décrit ci-dessous calcule, à partir d'une matrice F à coefficients dans $\mathbf{V}$ une matrice G en forme $\mathbf{k}$-échelonnée stricte telle que $\mathrm{Im}(G) = \mathrm{Sat}(\mathrm{Im}(F))$.
Cet algorithme est « incrémental » au sens suivant. Si l'on traite une matrice $[\,F_1 \mid F_2\,]$, on obtient une matrice $[\,G_1 \mid G_2\,]$ où G_1 est la matrice obtenue en traitant la matrice F_1.

Le lemme suivant nous sera utile dans la prochaine section.

Lemme 6 *Dans la procédure "Pivot de Gauss" décrite ci-dessus, si C est primitif et si l'indice $\mathrm{piv}(C)$ est distinct des $\mathrm{piv}(C^j)$ pour $j = 1, \dots, r$, alors le vecteur C' obtenu est primitif avec $\mathrm{piv}(C) = \mathrm{piv}(C')$.*

Démonstration. On considère l'affectation $C \leftarrow C - \frac{c_s}{c_{j,s}} C^j$ où $s = \mathrm{piv}(C^j)$ et c_s est le coefficient de C sur la ligne s. Posons $\ell = \mathrm{piv}(C)$. Le coefficient c_ℓ sur la ligne ℓ de C est remplacé par $c_\ell - \frac{c_s}{c_{j,s}} \cdot c_{j,\ell}$, où $c_{j,\ell}$ est le coefficient sur la ligne ℓ de C^j. Si $\ell > s$, c_s est résiduellement nul, si $\ell < s$ c'est $c_{j,\ell}$ qui est résiduellement nul, dans les deux cas le coefficient c_ℓ reste résiduellement inchangé. □

2 La V-saturation d'un $\mathbf{V}[X]$-module de type fini

Le travail que nous faisons dans cette section est un peu plus délicat et semble, étrangement, tout à fait nouveau. Il réalise en Calcul Formel un résultat théorique simple apparemment nouveau, et qui a fortiori n'avait jusqu'à maintenant aucune preuve constructive.

Théorème 7 *On se situe toujours dans le contexte 4. Si M est un sous-$\mathbf{V}[X]$-module de type fini de $\mathbf{V}[X]^n$ alors le $\mathbf{V}$-saturé de M dans $\mathbf{V}[X]^n$ est également un $\mathbf{V}[X]$-module de type fini.*

Remarque. Notons qu'en mathématiques classiques, tout domaine de valuation satisfait les hypothèses du contexte 4, par application du principe du tiers exclu. Notre preuve constructive du théorème 7 fournit donc aussi une preuve en mathématiques classiques sous la seule hypothèse que $\mathbf{V}$ est un domaine de valuation. La même remarque s'applique pour tous les résultats de cet article.

La démonstration du théorème résulte de la correction de l'algorithme qui calcule un système générateur fini du $\mathbf{V}$-saturé.

Vu comme $\mathbf{V}[X]$-module on a la base naturelle de $\mathbf{V}[X]^n$ notée $(\mathrm{f}_1, \dots, \mathrm{f}_n)$. Nous nous intéressons alors à une base naturelle de $\mathbf{V}[X]^n$ comme $\mathbf{V}$-module, qui est formée par les $\mathrm{e}_{i,k} = X^k\,\mathrm{f}_i$ avec l'ensemble d'indice $I = [\![1..n]\!] \times \mathbb{N}$. Nous munissons I de l'ordre lexicographique pour lequel

$$X^h\,\mathrm{f}_i < X^k\,\mathrm{f}_j \text{ si } i < j \text{ ou } i = j \text{ et } h < k.$$

Lorsque le module $\mathbf{V}[X]^n$ est vu comme un $\mathbf{V}$-module avec la base naturelle des $X^k\,\mathrm{f}_j$, nous parlons des « coordonnées » sur cette base. Lorsqu'il est vu comme un $\mathbf{V}[X]$-module avec la base naturelle des f_j, nous parlons des « coefficients » sur cette base.

On dispose au départ d'une liste $S = [s^1, \dots, s^m]$ de vecteurs dans $\mathbf{V}[X]^n$ qui forment un système générateur de M. On suppose sans perte de généralité que $m \geq 1$ et que les s^k sont non nuls. On note

$$E = \mathbf{V}\,s^1 + \cdots + \mathbf{V}\,s^m, \quad E_j = X^j\,E, \quad F'_k = \sum\nolimits_{j=0}^{k} E_j \quad \text{et} \quad G'_k = \mathrm{Sat}_{\mathbf{V},\mathbf{V}[X]^n}(F'_k).$$

On peut décrire F'_k et G'_k comme les modules images de deux matrices F_k et G_k. La matrice F_k est donnée, on la traite au moyen de l'algorithme de saturation de la section précédente, ce qui donne la matrice G_k.

La question qui se pose est de certifier qu'à partir d'un certain k, rien ne sert de continuer, car les éléments rajoutés dans la base de G'_k laissent inchangé le $\mathbf{V}[X]$-module engendré (notez que le $\mathbf{V}$-module $G'_k = \mathrm{Im}(G_k)$ grandit à chaque étape car $E \neq 0$).

Nous avons besoin de préciser quelques notations. Nous appelons « degré de E » et nous notons d le plus grand degré d'une des coordonnées de l'un des s^k. De la même manière, $d+k$ sera le degré de E_k ou celui de F_k. La matrice F_k peut donc être vue comme une matrice avec $n(1+d+k)$ lignes et $m(1+k)$ colonnes.

Si a est un vecteur $\mathbf{V}$-primitif de $\mathbf{V}[X]^n$, et si $\mathrm{piv}(a) = (j,r) \in I$ nous notons

$$\mathrm{index}(a) := j \text{ et } \mathrm{PrimMon}(a) := r.$$

L'entier $\mathrm{index}(a)$ est appelé l'*index de* a, l'entier $\mathrm{PrimMon}(a)$ son *premier exposant résiduel* et le couple $\mathrm{piv}(a)$ est *l'indice pivot de* a. Tout couple $(j,r) \in I$ sert d'indice pour un vecteur $X^r \mathrm{f}_j$ de la $\mathbf{V}$-base naturelle de $\mathbf{V}[X]^n$.

Notons H_k la matrice formée des colonnes que l'on rajoute à G_{k-1} pour obtenir la matrice G_k.

Fait 8 *Pour calculer la matrice G_{k+1} à partir de la matrice G_k, au lieu de traiter les générateurs de E_{k+1} (i.e. la liste $X^{k+1}S$), on peut se contenter de traiter les vecteurs colonnes de XH_k.*

Démonstration. Considérons la procédure simplifiée décrite ci-dessus.
Notons $\overline{G_k}$ les matrices successives obtenues par cette procédure simplifiée.
On vérifie sans difficulté par récurrence sur k que le $\mathbf{K}$-espace vectoriel engendré par les colonnes de $\overline{G_k}$ est le sous-espace $\mathbf{K}F'_k$ de $\mathbf{K}[X]^m$. En effet, toute colonne réduite à 0 est dans le $\mathbf{K}$-espace vectoriel engendré par les colonnes précédentes. Et toute colonne non réduite à 0, engendre, modulo les colonnes précédentes, une fois réduite, le même $\mathbf{K}$-espace vectoriel que la colonne qui lui donne naissance.
Les colonnes de $\overline{G_k}$ forment une base d'un sous-$\mathbf{V}$-module saturé de $\mathbf{V}[X]^m$, qui est donc égal à $\mathbf{K}F'_k \cap \mathbf{V}[X]^m$. Ceci montre que $\overline{G_k} = G_k$ □

Dans la suite, nous faisons référence à la procédure simplifiée, mais nous notons H_k et G_k au lieu de $\overline{H_k}$ et $\overline{G_k}$.

Aux matrices H_k et G_k nous associons plusieurs entiers :

- L'entier r_k est le nombre de colonnes de G_k, autrement dit le rang du $\mathbf{V}$-module libre $\mathrm{Im}(G_k)$.
- L'entier n_k, *nombre d'index pivots présents dans* H_k, est le cardinal de l'ensemble des $i \in [\![1..n]\!]$ tel qu'il existe une colonne C de H_k avec $\mathrm{index}(C) = i$. D'après le lemme 3, tous les index pivots présents dans H_{k-1} sont présents dans H_k, d'où l'on déduit par récurrence que n_k est aussi le nombre d'index pivots présents dans G_k. Donc la suite n_k est une suite croissante.
- L'entier u_k, *nombre de coordonnées disponibles pour* G_k *au vu de* n_k, est égal à $n_k(1+d+k)$. Si $n_{k+1} = n_k$ on a $u_{k+1} = u_k + n_k$.
- L'entier δ_k, *défaut de* H_k, est le nombre de colonnes C de H_k telles qu'il existe une autre colonne C' de H_k avec $\mathrm{index}(C) = \mathrm{index}(C')$ et $\mathrm{PrimMon}(C) < \mathrm{PrimMon}(C')$. Une telle colonne C sera dite *surnuméraire*. On a donc $r_k \leq u_k$ et $r_k = r_{k-1} + n_k + \delta_k$.
- L'entier $\Delta_k = u_{k+1} - r_k$ est la *place disponible à occuper à l'étape* $k+1$ *si* $n_k = n_{k+1}$.
  Si $n_k = n_{k+1}$, on a $\Delta_k = u_{k+1} - r_k = (u_k + n_k) - (r_{k-1} + n_k + \delta_k) = u_k - r_{k-1} - \delta_k = \Delta_{k-1} - \delta_k$.

Pour visualiser le défaut δ_k de H_k et la manière dont il évolue lorsque k augmente nous aurons recours, après la preuve, à des figures illustrant ce qui peut se passer. Mais peut-être la lecture de la preuve sera-t-elle facilité si on lit d'abord les commentaires qui accompagnent ces figures.

Le point essentiel est le suivant.

D'après le lemme 6, on est certain que lorsque l'on va traiter les colonnes successives de $H_{k+1} = XH_k$ au moyen de G_k, tout pivot (j,r) d'une colonne de H_k se retrouvera, décalé d'un cran, i.e. en position

$(j, r+1)$, comme pivot d'une colonne de H_{k+1}, sauf dans le cas où il s'agit d'un indice $(j, r+1)$ déjà présent dans G_0. Dans ce dernier cas, ou bien la collision réduit à 0 la colonne de XH_k (ce qui fait diminuer le défaut), ou bien un nouveau pivot est occupé par la colonne réduite (et rendue primitive). Ce nouveau pivot peut avoir deux effets distincts. Ou bien il se produit sur un index déjà occupé, et ne fait pas diminuer le défaut. Ou bien il se produit sur un index inoccupé, dans ce cas, le défaut diminue de 1 et le nombre n_k augmente entre n_k et n_{k+1}. On a donc établi la première affirmation du lemme suivant.

Lemme 9 *La suite δ_k est décroissante au sens large. Elle aboutit certainement à 0 pour k assez grand.*

Démonstration. On a déjà remarqué que si $n_{k+1} = n_k$ et $\delta_k > 0$ alors $\Delta_{k+1} < \Delta_k$. Pour un k assez grand on obtient donc $\delta_k = 0$ ou $n_{k+1} > n_k$. Dans le second cas, on reproduit la situation précédente. Comme la suite n_k est bornée par n, cela ne peut se produire qu'un nombre fini de fois. □

Lemme 10 *Si $\delta_k = 0$ le $\mathbf{V}[X]$-module engendré par G_k est le $\mathbf{V}$-saturé du $\mathbf{V}[X]$-module engendré par les s^j donnés au départ. On peut donc arrêter l'algorithme.*

Démonstration. Puisque la suite δ_k reste désormais nulle, il suffit de prouver que les colonnes de H_{k+1} sont dans le $\mathbf{V}[X]$-module engendré par les colonnes de G_k. Or vu le lemme 6, les colonnes de H_{k+1} sont dans le $\mathbf{V}$-module $\mathrm{Im}(G_k) + X\mathrm{Im}(H_k)$. □

Un exemple avec des figures.

La figure 1 représente les indices pivots de $G_0 = H_0$. Les 6 cercles blancs sont les éléments de $\mathrm{piv}(H_0)$.

Les cercles ou carrés noirs pleins correspondent à des éléments de la $\mathbf{V}$-base où aucun indice pivot de G_0 n'est présent. Les carrés noirs sont mis pour les index de pivots non présents : si toute une ligne est noire, on met des carrés pour insister.
Dans le cas présent on a donc $n = 5$, $d = 4$, $n_0 = 4$, $r_0 = 6$, $u_0 = 20$, $\Delta_0 = 14$, $\delta_0 = 2$.

On a entouré d'un grand cercle les deux éléments $\mathrm{piv}(H_0)$ des colonnes surnuméraires, qui correspondent à $\delta_0 = 2$.

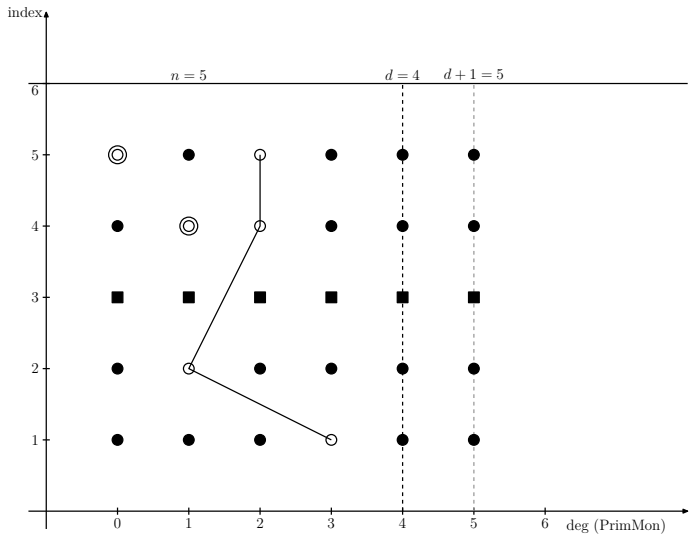

FIGURE 1 –

La ligne brisée noire joint les $\mathrm{piv}(C)$ pour les colonnes non surnuméraires de H_0.
D'après le lemme 6, on est certain que lorsque l'on va traiter les colonnes successives de $H_1 = XH_0$ au moyen de G_0, tout pivot (j, r) d'une colonne de H_0 se retrouvera, décalé d'un cran, i.e. en position

$(j, r+1)$ comme pivot d'une colonne de H_1, sauf dans le cas où il s'agit d'un indice $(j, r+1)$ déjà présent dans G_0.

Dans le cas de la figure 1, tous les pivots de H_0 à l'exception du pivot $(4,1)$ se retrouvent décalés d'un cran dans H_1. En particulier le pivot $(5,1)$ apparaîtra dans H_1, et l'on voit que ce sera pour une colonne surnuméraire (donc $\delta_1 \geq 1$).
Lorsqu'on va traiter la colonne XC telle que $\mathrm{piv}(C) = (4,1)$, une *collision* va se produire : un pivot de Gauss va être effectué pour réduire à 0 le coefficient en position $(4,2)$ de XC et la procédure de saturation va produire, ou bien le vecteur nul (auquel cas $\delta_1 = 1$ et H_1 n'aura que 5 colonnes, $r_1 = 11$), ou bien un vecteur C'' tel que $\mathrm{piv}(C'')$ remplisse une case non occupée dans l'espace a priori disponible (vecteurs de degré ≤ 5) avec nécessairement $\mathrm{piv}(C'') \notin \mathrm{piv}(G_0)$. Dans ce cas on aura $r_1 = 12$.

Nous allons maintenant examiner trois possibilités pour ce $\mathrm{piv}(C'')$, et nous donnons les 3 figures correspondantes pour H_1. Les pivots de H_0 seront des cercles (vides) gris et ceux de H_1 des cercles (vides) noirs. Les carrés ou ronds noirs pleins correspondent de nouveauà des cases vides qui pourraient a priori être remplies à l'avenir.

Dans le cas de la figure 2 on a $n_1 = 4$, $\delta_1 = 2$, $u_1 = 24$, $\Delta_1 = 12$. Lorsque l'on traitera XH_1 au moyen de G_1, des pivots en positions $(1,5)$, $(2,4)$, $(4,4)$ et $(5,4)$ seront produits dans H_2. Et deux collisions, respectivement en $(2,1)$ et $(5,2)$, donneront des résultats plus difficiles à prévoir. On pourra avoir $r_2 = 16$ avec $\delta_2 = 0$, ou $r_2 = 17$ (avec $\delta_2 = 1$ si $n_2 = 4$, ou $\delta_2 = 0$ si $n_2 = 5$), ou encore $r_2 = 18$.

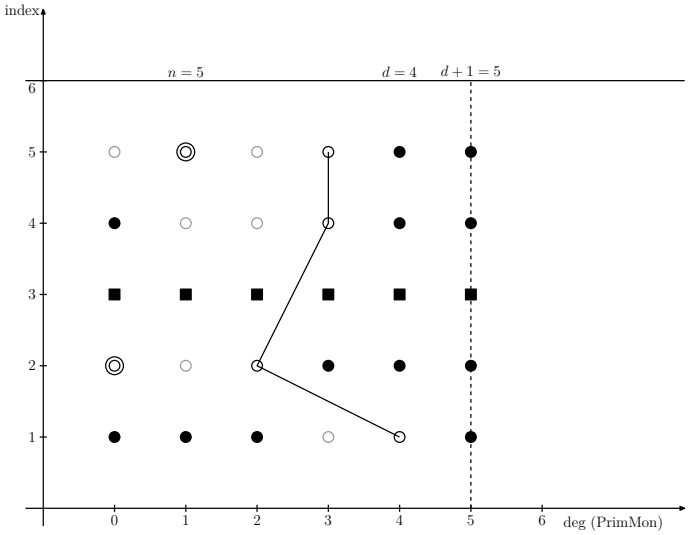

FIGURE 2 – Si la collision en $(4,2)$ produit un pivot en position $(2,0)$

Dans le cas de la figure 3 on a $n_1 = 5$, $\delta_1 = 1$, $u_1 = 30$, $\Delta_1 = 18$. Lorsque l'on traitera XH_1 au moyen de G_1, des pivots en positions $(1,5)$, $(2,3)$, $(3,4)$, $(4,4)$ et $(5,4)$ seront produits dans H_2. Et une collision, en $(5,2)$, donnera un résultat plus difficile à prévoir. Cela pourra réduire le vecteur à 0, auquel cas $\delta_2 = 0$, ou produire un nouveau vecteur, auquel cas $\delta_2 = 1$, car tous les index sont maintenant occupés par des pivots. Le nouveau vecteur aura a priori pour pivot n'importe lequel des carrés noirs indiqués sur la figure, ou aussi un pivot de degré 6.

Dans le cas de la figure 4 on a $n_1 = 4$ et $\delta_1 = 2$. Lorsque l'on traitera XH_1 au moyen de G_1, des pivots en positions $(1,5)$, $(2,4)$, $(4,4)$ et $(5,4)$ seront produits dans H_2. Et une collision, en $(5,2)$, donnera un résultat plus difficile à prévoir. La colonne surnuméraire de pivot $(2,2)$ ne produira a priori pas de collision, sauf dans le cas où la collision certaine, précédemment évoquée, est traitée avant et donne un vecteur réduit de pivot $(2,3)$.

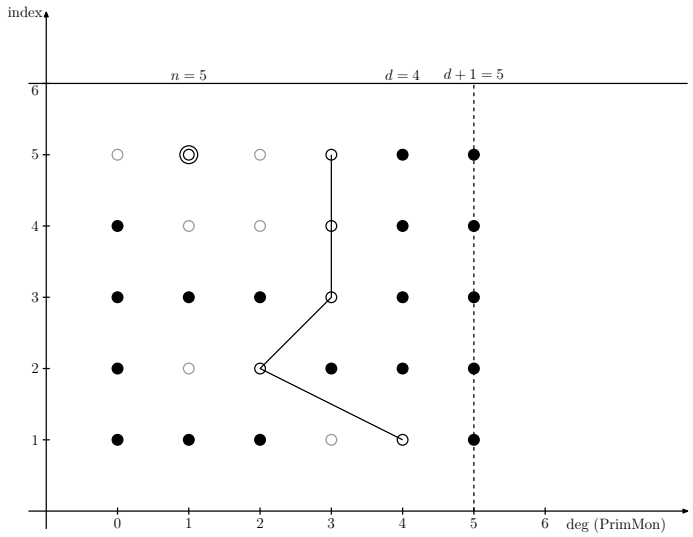

FIGURE 3 – Si la collision en $(4,2)$ produit un pivot en $(3,3)$

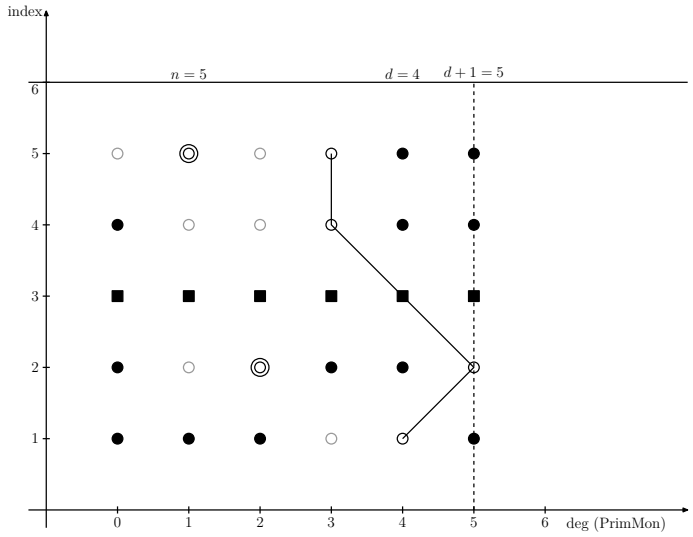

FIGURE 4 – Si la collision en $(4,2)$ produit un pivot en $(2,5)$

3 Module des syzygies pour un $\mathbf{V}[X]$-module de type fini

Théorème 11 *On se situe toujours dans le contexte 4. Soit $u_1,\dots,u_n \in \mathbf{V}[X]^k$ et $s^1,\dots,s^m \in \mathbf{K}[X]^n$ des générateurs du module des syzygies pour $(u_1,\dots,u_n)$ sur $\mathbf{K}[X]$. On peut supposer que les s^j sont dans $\mathbf{V}[X]^n$. Alors le module des syzygies pour $(u_1,\dots,u_n)$ sur $\mathbf{V}[X]$ est égal au $\mathbf{V}$-saturé dans $\mathbf{V}[X]^n$ du $\mathbf{V}[X]$-module engendré par $s^1,\dots,s^m$. En conséquence vu le théorème 7 ce module est de type fini. En particulier $\mathbf{V}[X]$ est un anneau cohérent.*

Démonstration. Un vecteur $f=(f_1,\dots,f_n)\in\mathbf{V}[X]^n$ tel que $\sum_j f_j u_j = 0$ s'écrit sous forme

$$f = a_1 s^1 + \cdots + a_m s^m$$

avec des $a_j \in \mathbf{K}[X]$. En multipliant cette relation par un $\alpha \neq 0$ convenable dans $\mathbf{V}$, on obtient les $\alpha\, a_i \in \mathbf{V}[X]$. Ceci montre que $\alpha\, f$ est dans le $\mathbf{V}[X]$-module engendré par les s^j. Et donc f est dans le $\mathbf{V}$-saturé du $\mathbf{V}[X]$-module engendré par les s^j. La réciproque est évidente. □

Ceci conduit à l'algorithme suivant pour calculer un système générateur fini du module des syzygies pour $u_1, \ldots, u_n \in \mathbf{V}[X]^k$ sur $\mathbf{V}[X]$:

1. Calculer des vecteurs $v^1, \ldots, v^m \in \mathbf{K}[X]^n$ qui forment un système générateur fini du module des syzygies pour $(u_1, \ldots, u_n)$ sur $\mathbf{K}[X]$.
2. En multipliant chaque v^j par un $\alpha_j \in \mathbf{K}$ convenable, le remplacer par un $s^j \in \mathbf{V}[X]^n$ primitif.
3. Calculer au moyen de l'algorithme de la section 2 un système générateur fini du $\mathbf{V}$-saturé du $\mathbf{V}[X]$-module $\langle s^1, \ldots, s^m \rangle$ dans $\mathbf{V}[X]^n$.

4 Annexe : des codes Magma

Nous présentons dans cette annexe des codes magma pour calculer le $\mathbf{V}$-saturé d'un sous-$\mathbf{V}[X]$-module de type fini d'un module $\mathbf{V}[X]^n$ (si $\mathbf{V}$ est un domaine de valuation résiduellement discret) en suivant la méthode décrite dans l'article.

Les commentaires sont ou bien sur une ligne, précédés de `//`, ou bien sur plusieurs lignes, entre les signes `/*` et `*/`.

```
%!TEX encoding =  UTF-8 Unicode

EchelonStrict := function(L, v0)
/*   L est une séquence strictement échelonnée de vecteurs (primitifs) de V[X]^n,
     v0 un vecteur de V[X]^n non nul
     On note V.L le sous-V-module de V[X]^n engendré par L.
     Il est saturé dans V[X]^n  car la séquence L est strictement échelonnée.
     La fonction retourne un vecteur v et un booléen.
     Le booléen est faux si v est dans V.L + V.v0, vrai sinon
     Si v0 est dans V.L, alors v = 0
     Sinon, v est primitif, la séquence L' = (L, v) est strictement échelonnée
     et on a Sat(V.L + V.v0) = V.L'
*/

v := v0 ;
for w in L do
   index, M, a := PivotAndCoefficient(w) ;
     // tuer le coefficient de v en position (index, M)
   v := v - a^(-1)*MonomialCoefficient(v[index], M)*w ;
    // si v est nul, c'est que v0 in L.V
   if v eq 0 then return v, false ; end if ;
end for ;

  // Ici, v est non nul. On le rend primitif
c := Contenu(v) ;   v := QuotientExact(v, c) ;
return v, not IsUnit(c) ;

end function ;

/*
S = s^1, ... , s^m est une séquence dans  V[X]^n
On considère dans V[X]^n le sous-V-module engendré par S , X*S , ... , X^k*S
et on note G'_k son saturé (comme V-module) (avec G'_k = {0} pour k < 0)
Dans la suite on calcule une V-base  G  de  G'_k  et  une liste  B  dans  V[X]^n
Cette liste  B,  extraite de  G,  suffira à engendrer, comme  V[X]-module,
le  V-saturé du V[X]-module engendré par  S.
*/
/*
Initialisations avec  G'_0
On calcule une V-base strictement échelonnée de G'_0
et le début de la liste  B  qui correspond à  G'_0
```

```
On initialise aussi  N,   qui est le nombre de vecteurs de  S
qui restent en vie après cette première réduction
*/

G := [Universe(S)| ] ; B:= G ;
// G est une liste vide d'objets du type de S
for v in S do
   w , new := EchelonStrict(G,v) ;
   if w ne 0 then Append(~G,w) ; end if ;
end for ;
B := G ; N := #G ;

/*  valeurs successives de G : V-base strictement échelonnée du V-module G'_k
    valeurs successives de B : G allégée de vecteurs inutiles du point de vue
                               du V[X]-moule engendré par  G'_k
    valeurs successives de N : dim_V (G'_k/G'_{k-1}) = nb de cols de H_k =
                               nombre de vecteurs de  X^k*S  qui restent en vie
                               après le calcul de G
*/

while true do

  // H <-- les N derniers vecteurs de G (= V-base de G'_{k-1}/G'_{k-2})
  H := G[#G-N+1 .. #G] ;

  // On calcule le défaut et on sort s'il est nul
  defaut := Defaut(H) ;

  if defaut eq 0 then break ; end if ;
  H := [X*v : v in H] ;
  N := 0 ;
  for v in H do
    w, new := EchelonStrict(G,v) ;
    if w ne 0 then Append(~G, w) ; N := N+1 ; end if ;
    if new then Append(~B, w) ; end if ;
  end for ;

end while ;
```

Références

[1] Adams W., Loustaunau P. *An Introduction to Gröbner Bases,* American Mathematical Society, (1994). 1

[2] Ducos L., Quitté C., Lombardi H., Salou M. *Théorie algorithmique des anneaux arithmétiques, de Prüfer et de Dedekind.* J. Algebra **281** (2004) 604–650.

[3] Glaz S. *Commutative Coherent Rings.* Lectures Notes in Math., vol. 1371, Springer Verlag, Berlin-Heidelberg-New York, second edition (1990). 1

[4] L. Gruson L., Raynaud M. *Critères de platitude et de projectivité. Techniques de "platification" d'un module.* Invent. Math. **13** (1971), 1–89 1

[5] Hadj Kacem A., Yengui I. *Dynamical Gröbner bases over Dedekind rings.* J. Algebra **324** (2010) 12-24. 1

[6] Lombardi H., Quitté C. *Algèbre Commutative, Méthodes Constructives.* À paraître. Version préliminaire en pdf disponible à l'url `http://hlombardi.free.fr/publis/LivresBrochures.html`.

[7] Lombardi H., Schuster P., Yengui I. *The Gröbner ring conjecture in one variable.* Math. Zeitschrift. **270**, (2012) 1181–1185. 1

[8] Mines R., Richman F., Ruitenburg W. *A Course in Constructive Algebra.* Universitext. Springer-Verlag, (1988). 1, 2

[9] Richman F. *Constructive aspects of Nœtherian rings.* Proc. Amer. Mat. Soc. **44** (1974), 436–441. 1

[10] Yengui I. *Dynamical Gröbner bases.* J. Algebra **301** (2006) 447–458. 1