Décomposition de Jordan-Chevalley-Dunford effective en temps linéaire

Jean-François Burnol, 13 avril 2017

Table des matières

Le titre est un peu abusif, mais cela est expliqué par la suite.

1 Rappel de trois algorithmes pour la décomposition de Dunford effective

Je résume ce qui figure dans mes fiches précédentes :

— http://jf.burnol.free.fr/agreg170406Dunford_NewAlgo.pdf

— http://jf.burnol.free.fr/agreg170408NewDun.pdf

— http://jf.burnol.free.fr/agreg170410NewtonSchroederDunford.pdf

On suppose donnée une matrice A à coefficients dans un corps $\mathbf{K}$, son polynôme caractéristique P_A et le polynôme unitaire $Q_A \in \overline{\mathbf{K}}[X]$ qui a les mêmes racines que P_A dans $\overline{\mathbf{K}}$ mais sans leurs multiplicités. En caractéristique nulle $Q_A \in \mathbf{K}[X]$ comme le montre la formule $Q_A = P_A/(P_A, P_A')$. En caractéristique positive, il est possible que Q_A ne soit pas défini sur $\mathbf{K}$. Certains algorithmes ci-dessous ne fonctionnent qu'en caractéristique nulle ou suffisamment grande, même si $Q_A \in \mathbf{K}[X]$. On suppose dorénavant (quitte à étendre $\mathbf{K}$ si la caractéristique est positive) que $Q_A \in \mathbf{K}[X]$.

Comme il a été démontré par Chevalley dans *Théorie des groupes de Lie, tome II* il est possible (c'est une variante moins détaillée de la décomposition de Jordan) d'écrire de manière unique $A = D+N$ avec $DN = ND$, N nilpotent et D diagonalisable sur $\overline{\mathbf{K}}$ car vérifiant $Q_A(D) = 0$. De plus D peut s'écrire comme un polynôme en A, et notre objectif est de calculer effectivement un tel polynôme.

Voici quelques algorithmes pour calculer un $D_N \in \mathbf{K}[X]$ (désolé pour le double emploi de N) convenable lorsque $P_A \in (Q_A^N)$ (on ne cherchait au début D_N que modulo P_A, mais tous les algorithmes nous en fournissent un qui marche pour Q_A^N et pas seulement pour P_A, alors dorénavant on se focalise sur Q_A^N). Dorénavant on ne parlera plus de P_A, on note Q pour Q_A, $q = \deg Q$, et H est l'unique polynôme de degré $< q$ avec $HQ' \equiv 1 \bmod Q$.

De plus pour estimer la complexité algorithmique, je fais des hypothèses simplificatrices :

— multiplier ou diviser deux polynômes de degrés $\mathcal{O}(d)$ coûte $\mathcal{O}(d^2)$,
— idem pour Euclide si on en a besoin, avec les coefficients de Bézout : $\mathcal{O}(d^2)$ pour trouver des coefficients de Bézout pour deux polynômes de degrés $\mathcal{O}(d)$,
— on néglige l'arithmétique des coefficients : soit on imagine que les calculs sont faits modulo un grand nombre premier, soit avec des nombres en virgule flottante. Si les coefficients sont des nombres rationnels manipulés exactement, les estimations de coût ne sont a priori plus correctes, et il faudrait pour affiner préciser si l'arithmétique haute précision fait de la multiplication sous-quadratique, etc… tout cela est trop compliqué pour que je puisse suivre, donc je trivialise cet aspect.

Algorithme de Chevalley (alias Newton) : on part de X et on itère la substitution $X \leftarrow X - HQ$; au bout de k itérations avec $2^k \geqslant N$, on a un D_N convenable. Pour que le coût ne soit pas gigantesque on travaille modulo Q^N (ou même modulo Q^{2^k} à la k^{e} étape).

En s'y prenant bien dans l'implémentation, je vois un coût de $\mathcal{O}(q^3 N^2 \log N)$.

Algorithme « des dérivées de $1/f$ » : on pose $U_1 = 1$ et on calcule itérativement dans $\mathbf{K}[X]$ les polynômes U_n avec $U_{n+1} = -nQ'U_n + QU_n'$, jusqu'à U_N. Alors $D_N = X + (N-1)\frac{U_{N-1}}{U_N} Q \bmod Q^N$, où la division par U_N est faite modulo Q^{N-1}.

C'est l'algorithme de

http://jf.burnol.free.fr/agreg170406Dunford_NewAlgo.pdf
http://jf.burnol.free.fr/agreg170408NewDun.pdf

J'estime en $\mathcal{O}(q^2 N^2)$.

Algorithme « $f\, d/df$ » : on définit itérativement $D_1 = X$, $D_2 = D_1 - HQD_1'$, $D_3 = D_2 - \frac{1}{2} HQD_2'$, …, $D_N = D_{N-1} - \frac{1}{N-1} HQD_{N-1}'$. À la fin on réduit modulo Q^N. C'est l'algorithme de

http://jf.burnol.free.fr/agreg170410NewtonSchroederDunford.pdf

J'estime en $\mathcal{O}(q^2 N^2)$.

En dernière page de http://jf.burnol.free.fr/agreg170410NewtonSchroederDunford.pdf j'explique que chacun des « chiffres en base Q » dans :

$$D_N = X - HQ + \frac{1}{2}\gamma_2 Q^2 - \cdots + \frac{(-1)^{N-1}}{(N-1)!}\gamma_{N-1}Q^{N-1} \pmod{Q^N}$$

peut s'obtenir en un coût $\mathcal{O}(q^2)$ donc un coût total $\mathcal{O}(q^2N)$ pour cette représentation, donnant l'unique solution modulo Q^N valable pour tous les endomorphismes annulés par Q^N.

Je n'écris pas ici les détails pratiques implémentant mes explications en petits caractères en page 14 de http://jf.burnol.free.fr/agreg170410NewtonSchroederDunford.pdf, préférant la formulation plus théorique suivante qui obtient le même résultat en donnant les formules « explicites » de la récurrence calculant « chiffre par chiffre » l'écriture de D_N en base Q.

2 Un algorithme à coût borné « par chiffre »

Théorème. *Soit $Q \in \mathbf{K}[X]$ unitaire à racines simples $\lambda_1, \ldots, \lambda_q$ dans $\overline{\mathbf{K}}$. On supposera $q \geqslant 2$.*

Soit $H \in \mathbf{K}[X]$ l'unique polynôme de degré au plus $q-1$ vérifiant $HQ' \equiv 1 \bmod Q$. Soit T avec

$$1 = HQ' + TQ$$

Posons $D_1 = \gamma_0 = X$ et $\alpha_1 = 1$, et définissons les $\alpha_n, \beta_n, \gamma_n$, $n \geqslant 1$, par les récurrences suivantes :

$$\begin{aligned} H\alpha_n &= \beta_n Q + \gamma_n \qquad (\textit{division euclidienne, } \deg \gamma_n < q) \\ \alpha_{n+1} &= \gamma_n' - n(T\alpha_n + Q'\beta_n) \end{aligned}$$

Alors, pour tout $n \geqslant 1$,

$$D_n = \sum_{k=0}^{n-1} \frac{(-1)^k}{k!}\gamma_k Q^k$$

est l'unique polynôme de degré $< nq$ réalisant les congruences

$$\begin{aligned} D_n &\equiv \lambda_1 \pmod{(X-\lambda_1)^n} \\ &\cdots \\ D_n &\equiv \lambda_q \pmod{(X-\lambda_q)^n} \end{aligned}$$

et donnant par conséquent la partie semi-simple de la multiplication par X modulo Q^n.

De plus les α_n vérifient $\deg \alpha_n \leqslant q-2$ *(et* $\deg \beta_n \leqslant q-3$*, les β_n sont nuls pour $q=2$), et peuvent aussi être calculés par la formule :*

$$\alpha_{n+1} = n\frac{Q'\gamma_n - \alpha_n}{Q} + \gamma_n'$$

La relation suivante est vérifiée :

$$D_n' = \frac{(-1)^{n-1}}{(n-1)!}\alpha_n Q^{n-1}$$

Preuve. C'est la relation entre D_n' et α_n qui est cruciale. On a bien $D_1' = 1 = \alpha_1 Q^0$. Supposons $D_n' = \frac{(-1)^{n-1}}{(n-1)!}\alpha_n Q^{n-1}$ et évaluons D_{n+1}'.

$$\begin{aligned}
D_{n+1} &= D_n + \frac{(-1)^n}{n!}\gamma_n Q^n \\
D_{n+1}' &= \frac{(-1)^{n-1}}{(n-1)!}\alpha_n Q^{n-1} + \frac{(-1)^n}{n!}\gamma_n' Q^n + \frac{(-1)^n}{n!}\gamma_n n Q^{n-1} Q' \\
&= \frac{(-1)^{n-1}}{(n-1)!}\alpha_n (HQ' + TQ) Q^{n-1} + \frac{(-1)^n}{n!}\gamma_n' Q^n + \frac{(-1)^n}{n!}\gamma_n n Q^{n-1} Q' \\
&= \frac{(-1)^n}{n!} Q^n \cdot \left(-nT\alpha_n + \gamma_n'\right) + \frac{(-1)^n}{n!} Q^{n-1} Q' \cdot \left(-nH\alpha_n + n\gamma_n\right) \\
&= \frac{(-1)^n}{n!} Q^n \cdot \left(-nT\alpha_n + \gamma_n'\right) + \frac{(-1)^n}{n!} Q^{n-1} Q' \cdot \left(-nQ\beta_n\right) \\
&= \frac{(-1)^n}{n!} Q^n \cdot \left(-nT\alpha_n + \gamma_n' - nQ'\beta_n\right) \\
&= \frac{(-1)^n}{n!} Q^n \alpha_{n+1}
\end{aligned}$$

On en déduit que en λ_j, le polynôme dérivé D_{n+1}' a un zéro d'ordre au moins n. Donc $D_{n+1} - D_{n+1}(\lambda_j)$ y a un zéro d'ordre au moins $n+1$. Mais $D_{n+1}(\lambda_j) = D_n(\lambda_j) = D_1(\lambda_j) = \lambda_j$. Le polynôme D_{n+1} résout donc le problème Chinois d'ordre $n+1$. Et par construction, puisque $\deg \gamma_n < \deg Q = q$, chaque D_n est de degré $< n \deg Q$. Donc les D_n sont bien les uniques polynômes de degrés $< n \deg Q$ solutions des congruences à l'ordre n.

Par conséquent D_n' est de degré au plus $nq-2$, or il est divisible par Q^{n-1} qui est de degré $(n-1)q$ (*lorsque* $n-1>0$!), le quotient α_n est donc de degré au plus $q-2$ lorsque $n \geqslant 2$. Comme on a supposé $q \geqslant 2$ et que $\alpha_1 = 1$, c'est en fait valable aussi pour $n=1$.

On voit aussi, comme $\deg H \leqslant q-1$, que $\deg \beta_n \leqslant q-2+q-1-q = q-3$.

Il ne reste plus qu'à établir

$$\alpha_{n+1} = n\frac{Q'\gamma_n - \alpha_n}{Q} + \gamma_n'$$

ce que nous faisons ainsi :

$$\begin{aligned} Q\alpha_{n+1} &= Q\cdot(\gamma_n' - n(T\alpha_n + Q'\beta_n)) \\ &= Q\gamma_n' - n(1-HQ')\alpha_n - nQ'(H\alpha_n - \gamma_n) \\ &= Q\gamma_n' - n(\alpha_n - Q'\gamma_n) \end{aligned}$$

La preuve est complète. □

Remarque 1 : lorsque $q=1$, $Q=X-\lambda$, $H=1$, l'algorithme fonctionne aussi et donne $D_1=X$ et $D_2=D_3=\cdots=\lambda$, $\alpha_n=0$ pour $n\geqslant 2$, $\gamma_1=1$, $\gamma_n=0$ pour $n\geqslant 2$, mais $D_n = X-1\cdot(X-\lambda)$ n'est pas une écriture « en base Q » car $\deg X=\deg Q$. Du coup, subtilement certains passages de la preuve ci-dessus ont des problèmes et pour simplifier j'ai pris $q\geqslant 2$ tout du long.

Remarque 2 : je laisse l'explicitation du cas $q=2$ en exercice…

Remarque 3 : chaque étape à un coût $\mathcal{O}(q^2)$ puisque les α_n sont de degrés au plus $q-2$.

3 Fractions rationnelles

On peut faire une variante « sans réduction modulo Q ».

Théorème. *Soit à nouveau $Q\in\mathbf{K}[X]$ comme précédemment. Soit H une fraction rationnelle telle que $\frac{1}{Q'}-H$ s'annule aux λ_j. On pose $T=(1-HQ')/Q$ de sorte que*

$$1 = HQ' + TQ$$

où les fractions rationnelles H et T sont sans pôles aux λ_j, $1\leqslant j\leqslant q$.

Définissons par récurrence $A_1=1$ puis les fractions rationnelles

$$A_{n+1} = (HA_n)' - nTA_n$$

qui sont sans singularités aux λ_j, $1\leqslant j\leqslant q$.

Alors la fraction rationnelle

$$F_n = X - HA_1Q + \sum_{k=2}^{n-1}\frac{(-1)^k}{k!}HA_kQ^k$$

est définie aux λ_j et résout le problème chinois correspondant au problème de la décomposition de Dunford. *Par ailleurs*

$$F_n' = \frac{(-1)^{n-1}}{(n-1)!} A_n Q^{n-1} \quad \text{et} \quad F_{n+1} = (1 - \frac{1}{n} HQ \frac{d}{dX}) F_n$$

Preuve. Les A_n n'ont pas de pôles en les λ_j et on vérifie sans peine le télescopage donnant

$$F_n' = \frac{(-1)^{n-1}}{(n-1)!} A_n Q^{n-1}$$

d'où le fait que F_n soit, en tant que fraction rationnelle, une solution au problème chinois (en tenant compte du fait que $F_n(\lambda_j) = \lambda_j$ pour tout j). Et aussi $HQF_n' = -n\frac{(-1)^n}{n!}(HA_n)Q^n$, d'où la relation de récurrence $F_{n+1} = F_n - \frac{1}{n} HQF_n'$ de l'énoncé. □

On peut aussi varier les H à chaque cran, mais bon.

On peut revenir à l'Analyse, mais bon...

4 Un exemple avec Maple

Je me suis aperçu que je ne savais pas obtenir en une seule opération le quotient et le reste dans la division euclidienne avec Maple...

Bref, voici maintenant, l'exemple avec $Q = (X-1)(X-2)(X-3)$ et on va jusqu'à $N = 5$. Je vérifie à la fin que tout marche bien en ce qui concerne les congruences chinoises pour $N = 4$ et $N = 5$.

J'utilise AN, BN, GN pour α_n, β_n, γ_n.

```
> restart; Q := (X-1)*(X-2)*(X-3);
```

$$Q := (X-1)(X-2)(X-3)$$

```
> Q := expand(Q);
```

$$Q := X^3 - 6X^2 + 11X - 6$$

```
> Qprime := diff(Q, X);
```

$$Qprime := 3X^2 - 12X + 11$$

```
> gcdex(Qprime, Q, X, 'H', 'T');
```

$$1$$

```
> H, T, expand(H*Qprime+T*Q);
```

$$5 - 6X + 3/2\,X^2,\ 9 - 9/2\,X,\ 1$$

```
> A1 := 1;
```

$$A1 := 1$$

```
> P1 := H*A1:
```

```
> G1 := rem(P1, Q, X);
```

$$\boxed{G1 := 5 - 6X + 3/2X^2}$$

```
> B1 := quo(P1, Q, X);
```

$$B1 := 0$$

```
> A2 := diff(G1, X)-T*A1-Qprime*B1;
```

$$A2 := 15/2X - 15$$

```
> P2 := H*A2:
```

```
> G2 := rem(P2, Q, X);
```

$$\boxed{G2 := \frac{15}{4}X - 15/2}$$

```
> B2 := quo(P2, Q, X);
```

$$B2 := \frac{45}{4}$$

```
> A3 := diff(G2, X)-2*(T*A2+Qprime*B2);
```

$$A3 := -\frac{975}{4} - 2\,(9 - 9/2X)(15/2X - 15) - \frac{135}{2}X^2 + 270X$$

```
> A3 := expand(A3);
```

$$A3 := \frac{105}{4}$$

```
> P3 := H*A3:
```

```
> G3 := rem(P3, Q, X);
```

$$\boxed{G3 := \frac{525}{4} - \frac{315}{2}X + \frac{315}{8}X^2}$$

```
> B3 := quo(P3, Q, X);
```

$$B3 := 0$$

```
> A4 := diff(G3, X)-3*(T*A3+Qprime*B3);
```

$$A4 := -\frac{3465}{4} + \frac{3465}{8}X$$

```
> P4 := H*A4;
```

$$P4 := \left(5 - 6X + 3/2X^2\right)\left(-\frac{3465}{4} + \frac{3465}{8}X\right)$$

```
> G4 := rem(P4, Q, X);
```

$$\boxed{G4 := \frac{3465}{16}X - \frac{3465}{8}}$$

```
> B4 := quo(P4, Q, X);
```

$$B4 := \frac{10395}{16}$$

```
> A5 := diff(G4, X)-4*(T*A4+Qprime*B4);
```

$$A5 := -\frac{453915}{16} - 4\,(9 - 9/2\,X)\left(-\frac{3465}{4} + \frac{3465}{8}X\right) - \frac{31185}{4}X^2 + 31185\,X$$

```
> A5 := expand(A5);
```

$$A5 := \frac{45045}{16}$$

```
> D5 := X-G1*Q+(1/2)*G2*Q^2-(1/6)*G3*Q^3+(1/24)*G4*Q^4:

> rem(D5, (X-1)^5, X);
```

$$1$$

```
> rem(D5, (X-2)^5, X);
```

$$2$$

```
> rem(D5, (X-3)^5, X);
```

$$3$$

```
> simplify((diff(D5, X))/Q^4);
```

$$\frac{15015}{128}$$

```
> (1/24)*A5;
```

$$\frac{15015}{128}$$

```
> D4 := X-G1*Q+(1/2)*G2*Q^2-(1/6)*G3*Q^3:

> simplify((diff(D4, X))/Q^3+(1/6)*A4);
```

$$0$$

Donc, tout marche bien.

Si l'on regarde attentivement on constate que les γ_{2n} sont multiples de T et les γ_{2n+1} multiples de H (je laisse en exercice les facteurs en jeu)! ceci m'a plongé dans des abîmes de perplexité, jusqu'à ce que je réalise que *pour* $(X-1)(X-2)(X-3)$ *mais pas en général pour* $\deg Q = 3$, H' est proportionnel à T! ($H' = -\frac{2}{3}T$). C'est un machin particulier qui se réalise pour ce polynôme en particulier! **OUF**, je ne vais pas avoir à prolonger éternellement ces histoires. Enfin, j'espère…