Euclide et Fibonacci

Jean-François Burnol, 31 mars 2011

1 Euclide et matrices 2 × 2

L'algorithme d'Euclide traditionnel calcule $(a\,,b)$ pour $a \geq b > 0$.[1] Posons soigneusement les notations : $r_0 = a$, $r_1 = b$, $a = q_1 b + r_2$, $b = q_2 r_2 + r_3$, $r_2 = q_3 r_3 + r_4$, ... La première étape fait la division euclidienne de a par b. Le quotient est $q_1 \geq 1$, le reste $r_2 < r_1$. La deuxième étape a un quotient $q_2 \geq 1$, un reste $r_3 < r_2$. La n^e étape est avec un quotient $q_n \geq 1$ et un reste $r_{n+1} < r_n$. Les restes sont tous positifs ou nuls, et même strictement positifs sauf le dernier. Les quotients sont tous au moins égaux à 1.

[A] $r_{n-1} = q_n r_n + r_{n+1}\,, \quad 1 \leq q_n\,, \quad 0 \leq r_{n+1} < r_n \qquad (n^e \text{ étape})$

La suite des restes est strictement décroissante à partir de $n = 1$ (mais on a autorisé $r_0 = r_1$). L'algorithme s'arrête lorsque r_{n+1} est nul. On dit qu'il y a eu alors $N = n$ étapes. Par exemple, pour calculer $(10\,,5)$ j'ai besoin d'une étape : $10 = 2\times 5 + 0$, $N = 1$. Pour calculer $(11\,,5)$ j'ai besoin de deux étapes : $11 = 2\times 5 + 1$, $5 = 5\times 1 + 0$, $N = 2$. Bien sûr dans la pratique on est tenté de dire qu'on a fini dès la première étape. Mais pour le théoricien modélisateur, il ne faut pas s'autoriser ce raccourci. Non, encore lui faut-il faire la division par 1 pour s'assurer que le reste est 0. Imaginez-le comme un comptable très pointilleux. Sinon, lui et nous allons nous embrouiller dans le décompte du nombre d'étapes. Le PGCD est le dernier reste non nul. C'est r_N lorsqu'il y a eu N étape. Par exemple si b divise a, il y a 1 étape, et le PGCD est $(a, b) = b = r_1$. Si $a = b + 1$, il y a 2 étapes,[2] et $(a, b) = 1 = r_2$.

Il est extrêmement naturel[3] d'associer à la récurrence **[A]** une interprétation matricielle. La relation **[A]** s'écrit :

$$\begin{pmatrix} r_n \\ r_{n-1} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & q_n \end{pmatrix} \begin{pmatrix} r_{n+1} \\ r_n \end{pmatrix}$$

Et au final toutes les étapes sont rassemblées en :

[B] $$\begin{pmatrix} b \\ a \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & q_1 \end{pmatrix} \cdots \begin{pmatrix} 0 & 1 \\ 1 & q_N \end{pmatrix} \begin{pmatrix} 0 \\ d \end{pmatrix} \quad \text{avec } d = (a\,,b) = r_N$$

1. on utilise parfois la notation $a \wedge b$, mais comme je n'arrive jamais à me rappeler si c'est $a \wedge b$ ou $a \vee b$, je préfère $(a\,,b)$. Notez la subtile différence typographique entre le couple (a, b) et le pgcd $(a\,,b)$. Et si l'on veut une notation pour le ppcm, pourquoi pas alors $[a\,,b]$. Bien sûr on préférera au collège, dans un premier temps $\mathrm{pgcd}(a, b)$ et $\mathrm{ppcm}(a, b)$.

2. sauf si $b = 1$...

3. on pourra penser au lien entres les matrices 2×2 et les homographies, au lien entre les homographies et les fractions continues, au lien entre les fractions continues pour un nombre rationnel et la division euclidienne.

J'ai noté d le pgcd de a et de b, comme il est traditionnel.

2 Une subtilité subtile [4]

Réciproquement partons d'entiers N, d, $q_1, \dots q_N$ tous au moins égaux à 1 et formons :

$$\begin{pmatrix} b \\ a \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & q_1 \end{pmatrix} \cdots \begin{pmatrix} 0 & 1 \\ 1 & q_N \end{pmatrix} \begin{pmatrix} 0 \\ d \end{pmatrix}$$

Ceci correspond-t-il à l'algorithme d'Euclide partant de (a, b) ? Pour cela, supposons $0 \leq C < B$, $q \geq 1$ et formons :

$$\begin{pmatrix} B \\ A \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & q \end{pmatrix} \begin{pmatrix} C \\ B \end{pmatrix}$$

Donc $A = qB + C$, et C est le reste de la division euclidienne de A par B, ok. Mais peut-on continuer avec un autre q et (B, A) à la place de (C, B) ? pour cela il faut regarder si $B < A$. C'est vrai avec une unique exception : **$q = 1$ et $C = 0$**. Notre théorème est donc le suivant :

Théorème 1. *Il y a une correspondance biunivoque entre le déroulement de l'algorithme d'Euclide pour les couples* (a, b) *avec* $a > b > 0$ *et les écritures matricielles*

[C] $$\begin{pmatrix} b \\ a \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & q_1 \end{pmatrix} \cdots \begin{pmatrix} 0 & 1 \\ 1 & q_N \end{pmatrix} \begin{pmatrix} 0 \\ d \end{pmatrix}$$

avec $d \geq 1$, $N \geq 1$, $q_1, \dots, q_{N-1} \geq 1$ *et* $\mathbf{q_N \geq 2}$. *Si l'on veut autoriser* $a = b > 0$ *alors il faut dire plutôt : avec* $N \geq 1$, $q_1, \dots, q_N \geq 1$, *et* $q_N \geq 2$ *si* $N \geq 2$.

Dans la suite on aura $a > b$, car comme on vient de le voir cela simplifie la discussion.

3 Bezout

Écrivons :

[D] $$\begin{pmatrix} u_N & \beta_N \\ v_N & \alpha_N \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & q_1 \end{pmatrix} \cdots \begin{pmatrix} 0 & 1 \\ 1 & q_N \end{pmatrix}$$

De **[C]** il vient :

$$\beta_N = \frac{b}{d} \quad \alpha_N = \frac{a}{d}$$

4. pour ne pas dire pénible…

Le déterminant est multiplicatif, donc

$$u_N\alpha_N - v_N\beta_N = (-1)^N$$

On a ainsi une identité de Bezout :

[E] $u_N a - v_N b = (-1)^N d$

Traditionnellement on obtient cela en « remontant » les calculs. Avec nos matrices cela veut dire qu'on est passé aux inverses :

$$\begin{pmatrix}0\\d\end{pmatrix} = \begin{pmatrix}0 & 1\\1 & q_N\end{pmatrix}^{-1}\cdots\begin{pmatrix}0 & 1\\1 & q_1\end{pmatrix}^{-1}\begin{pmatrix}b\\a\end{pmatrix} = \begin{pmatrix}-q_N & 1\\1 & 0\end{pmatrix}\cdots\begin{pmatrix}-q_1 & 1\\1 & 0\end{pmatrix}\begin{pmatrix}b\\a\end{pmatrix}$$

En effet, « remonter » les calculs signifient concrètement faire exactement ce qui permettrait de calculer le produit des matrices à droites de l'équation ci-dessus (en allant de $n = N$ vers $n = 1$). Donc en fait c'est comme si on calculait l'inverse de la matrice totale, qui n'est autre que

$$(-1)^N\begin{pmatrix}\alpha_N & -\beta_N\\-v_N & u_N\end{pmatrix}$$

« Remonter » les calculs mène ainsi à la même relation **[E]** justifiée ci-dessus via un calcul de déterminant, $d = (-1)^N u_N a - (-1)^N v_N b$. Comme « remonter » (pardon pour les répétitions…) signifie concrètement faire des opérations avec des soustractions (comme il est clair vu les matrices ci-dessus), ce qui donne facilement des erreurs de signe, il est certainement bien plus sûr de procéder plutôt via le calcul du produit matriciel :

$$\begin{pmatrix}0 & 1\\1 & q_1\end{pmatrix}\cdots\begin{pmatrix}0 & 1\\1 & q_N\end{pmatrix}$$

On peut soit le faire de gauche à droite, soit de droite à gauche ; dans le premier cas on fait des combinaisons de colonnes :

$$\begin{pmatrix}0 & 1\\1 & q_1\end{pmatrix}\to\begin{pmatrix}1 & 0+q_2\times 1\\q_1 & 1+q_2\times q_1\end{pmatrix}\to\begin{pmatrix}0+q_2\times 1 & 1+q_3\times(0+q_2\times 1)\\1+q_2\times q_1 & q_1+q_3\times(1+q_2\times q_1)\end{pmatrix}\to\cdots$$

Ou alors, et c'est peut-être plus simple (ça dépend du cerveau de chacun, pour une machine c'est la même chose), on calcule de droite à gauche, par des combinaisons de lignes donc. Je vous laisse écrire le truc, ou mieux, faire un calcul concret. Cette méthode par combinaison de lignes descend de $n = N$ à $n = 1$, comme l'on fait lorsque l'on « remonte » les divisions euclidiennes ; la grande différence c'est que l'on n'a *aucun* signe, donc moins de risques d'erreurs.

4 Application : construire un exercice

Exemple : $N = 4$, $q_1 = 7$, $q_2 = 4$, $q_3 = 2$, $q_4 = 5$. Je calcule à partir de la droite par des combinaisons de lignes :

$$\begin{pmatrix} 9 & 49 \\ 65 & 354 \end{pmatrix} \leftarrow \begin{pmatrix} 2 & 11 \\ 9 & 49 \end{pmatrix} \leftarrow \begin{pmatrix} 1 & 5 \\ 2 & 11 \end{pmatrix} \leftarrow \begin{pmatrix} 0 & 1 \\ 1 & 5 \end{pmatrix}$$

Cela nous donne une matrice de déterminant 1. Et vous savez qu'en donnant à vos élèves : *calculer le* PGCD *de* 343 *(*7×49*) et* 2478 *(*7×354*)*, ils trouveront 7 après 4 étapes, dont les quotients successifs seront $q_1 = 7$, $q_2 = 4$, $q_3 = 2$, $q_4 = 5$. De plus vous connaissez à l'avance l'identité de Bezout

$$9 \times 2478 - 65 \times 343 = 7$$

qu'ils pourront (peut-être) retrouver en « remontant » les calculs ! Il faut absolument encourager les calculs, et donc l'exercice terminera en apothéose lorsqu'ils vérifieront à la main que cette identité fonctionne.

Je répète pour ceux à qui on a donné 2478 et 343 qu'une fois trouvés les quotients $q_1 = 7$, $q_2 = 4$, $q_3 = 2$, $q_4 = 5$, la manière la plus simple d'aboutir à l'identité de Bezout est de calculer le produit matriciel :

$$\begin{pmatrix} 0 & 1 \\ 1 & 7 \end{pmatrix}\begin{pmatrix} 0 & 1 \\ 1 & 4 \end{pmatrix}\begin{pmatrix} 0 & 1 \\ 1 & 2 \end{pmatrix}\begin{pmatrix} 0 & 1 \\ 1 & 5 \end{pmatrix}$$

car cela ne contient aucun signe. Les coefficients de l'identité de Bezout se liront dans la première colonne, l'identité est que le déterminant vaut ± 1 (ici 1).

5 Nombre d'étapes et Fibonacci

Comme $q_1 \geq 1, \ldots, q_N \geq 2$, il est vrai et aussi facile de montrer par récurrence :

$$\begin{pmatrix} u_N & \beta_N \\ v_N & \alpha_N \end{pmatrix} = \overbrace{\begin{pmatrix} 0 & 1 \\ 1 & q_1 \end{pmatrix} \cdots \begin{pmatrix} 0 & 1 \\ 1 & q_{N-1} \end{pmatrix}}^{N-1 \text{ termes}} \begin{pmatrix} 0 & 1 \\ 1 & q_N \end{pmatrix} \geq \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}^{N-1} \begin{pmatrix} 0 & 1 \\ 1 & 2 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}^{N+1}$$

au sens où tous les coefficients de la matrice de gauche sont au moins égaux à ceux de la matrice de droite (ne pas confondre cela avec la notation $M \geq N$ pour les matrices symétriques).

Or nous connaissons cette matrice de droite, et si nous ne la connaissons pas il est très facile de découvrir qu'elle vaut, avec les nombres de Fibonacci, $F_0 = 0$, $F_1 = 1$, $F_{n+2} = F_n + F_{n+1}$:

$$\begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}^{N+1} = \begin{pmatrix} F_N & F_{N+1} \\ F_{N+1} & F_{N+2} \end{pmatrix}$$

Ainsi :

[F] $u_N \geq F_N, \quad v_N \geq F_{N+1}, \quad \beta_N \geq F_{N+1}, \quad \alpha_N \geq F_{N+2}$

Mais nous savons que :

$$\alpha_N = \frac{a}{d} \leq a, \qquad \beta_N = \frac{b}{d} \leq b$$

Cela nous donne le Théorème suivant :

Théorème 2. *Le nombre* N *d'étapes qui sont nécessaires pour calculer le* PGCD *de* a *et de* b *(*$0 < b < a$*) par l'algorithme d'Euclide est majoré par les inégalités* $F_{N+1} \leq b$ *et* $F_{N+2} \leq a$. *Ainsi pour estimer* N *on calcule* k *tel que* $F_k \leq b < F_{k+1}$. *Alors* $N \leq k - 1$. *Mais si* a *est lui aussi* $< F_{k+1}$ *alors en fait* $N \leq k - 2$.

Comme on a la formule explicite :

$$F_k = \frac{\alpha^k - \beta^k}{\alpha - \beta}, \qquad \alpha = \Phi = \frac{1 + \sqrt{5}}{2}, \quad \beta = -\alpha^{-1} = \frac{1 - \sqrt{5}}{2} = 1 - \Phi, \quad -1 < \beta < 0\ ,$$

on voit que k est contraint par

$$\ln(\alpha^k - \beta^k) \leq \ln(b) + \ln(\alpha - \beta)$$

$$k \ln(\alpha) \leq \ln(b) + \ln(\alpha - \beta) - \ln\left(1 - (\beta/\alpha)^k\right)$$

Bien sûr j'ai supposé k au moins égal à 1 pour éviter d'avoir ln(0), et comme $b \geq 1 = F_2$, en fait on peut même supposer $k \geq 2$ dans ce qui suit qui recherche le plus grand k possible.

Comme β/α est négatif (et de valeur absolue inférieure à 1), la valeur maximale de

$$-\ln\left(1 - (\beta/\alpha)^k\right)$$

est obtenue pour k pair le plus petit possible, donc $k = 2$. Donc :

$$k \ln(\alpha) \leq \ln(b) + \ln(\alpha - \beta) - \ln(1 - (\beta/\alpha)^2) = \ln(b) + \ln\left(\alpha^2 \frac{1}{\alpha + \beta}\right) = \ln(b) + 2\ln(\alpha)$$

Comme notre estimation du nombre N d'étapes est $N \leq k - 1$ nous obtenons :

Théorème 3. *Le nombre* N *d'étapes est majoré par* $\frac{\ln b}{\ln \Phi} + 1$ *et aussi par* $\frac{\ln a}{\ln \Phi}$ *avec* Φ *le nombre d'or.*

La deuxième estimation est meilleure que la première s'il existe un entier x avec $\frac{\ln a}{\ln \Phi} < x \leq \frac{\ln b}{\ln \Phi} + 1$. On aurait à la fois

$$a < \Phi^x \qquad \text{et} \qquad \Phi^{x-1} \leq b$$

Ainsi s'il existe un entier x avec $\Phi^{x-1} \leq b < a < \Phi^x$ l'estimation du nombre d'étapes qui utilise b donne $N \leq x$ mais celle qui utilise a donne $N \leq x - 1$ soit 1 étape en moins. Donc utiliser la formule avec a ne peut nous faire gagner que 1 et en général nous fait plutôt perdre, puisque a n'est pas limité supérieurement. De toute façon le véritable nombre d'étapes peut être bien inférieur encore.

Comme en fait F_k est proche de $\sqrt{\frac{1}{5}}\Phi^k$, la méthode plus exacte avec les nombres de Fibonacci donnerait quelque chose approximativement comme $N \lesssim \frac{\ln b}{\ln \Phi} + \frac{\ln 5^{1/2}}{\ln \Phi} - 1 = \frac{\ln b}{\ln \Phi} + 0.6722759\ldots$, donc au mieux on pourrait gagner 1 dans l'estimation de N. En pratique, j'imagine qu'en règle général l'estimation est de toute façon largement supérieure à la réalité, même si à vrai dire je n'en sais rien. J'imagine que cette question a été étudiée (et que ça ne doit pas être tout-à-fait trivial…).

En tout cas elle montre que la majoration $N \leq b$ provenant de $r_1 = b > r_2 > \cdots > r_N = d > 0$ est extrêmement mauvaise : car calculer le PGCD de nombres avec 8 chiffres dans leur écriture décimal ne coûte, en fait, pas bien plus cher que deux fois le nombre d'étapes pour les pires des cas avec 4 chiffres.

6 Division euclidienne modifiée

La division euclidienne modifiée de a par b, avec $a > b > 0$ consiste à remplacer le reste r par $b - r$ lorsque $r > \frac{1}{2}b$, de sorte qu'au lieu d'avoir $a = bq + r$, on a $a = b(q + 1) - (b - r)$, donc cela met la division euclidienne sous la forme $a = bq' + \varepsilon r'$, avec $\varepsilon = \pm 1$, $0 \leq r' \leq \frac{1}{2}b$, et q' non plus la partie entière de $\frac{a}{b}$ mais l'entier le plus proche de $\frac{a}{b}$; lorsque la fraction est à mi-chemin entre deux entiers consécutifs, on choisit celui du dessous (plus haut j'ai écrit « si $r > \frac{1}{2}b$ alors $r \mapsto b - r$, $q \mapsto q + 1$ »). Au niveau matriciel c'est donc qu'on a plutôt maintenant :

$$\begin{pmatrix} b \\ a \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ \varepsilon & q' \end{pmatrix} \begin{pmatrix} r' \\ b \end{pmatrix} \qquad \begin{pmatrix} 0 & 1 \\ 1 & q \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -1 & q + 1 \end{pmatrix} \begin{pmatrix} -1 & 1 \\ 0 & 1 \end{pmatrix}$$

Bref, à vue de nez c'est plus dur de dire des choses intéressantes ici avec les matrices, alors que trivialement comme $r'_2 \leq \frac{1}{2}b$, on a $r'_{N+1} = 0$ dès que $2^{-N}b < 1$, ou encore dès que $b < 2^N$. Le nombre d'étapes est donc au plus $\left[\frac{\ln b}{\ln 2}\right] + 1 \leq \frac{\ln b}{\ln 2} + 1$. Dans le cas de deux nombres de Fibonacci successifs F_{k+1} et F_{k+2}, au lieu de k étapes, on s'attend donc à au plus de l'ordre de $\frac{\ln \Phi}{\ln 2}k \simeq 0.7 \cdot k$ étapes. Soit un gain de 30%. Mais en réalité c'est 50%, car **Exercice** : l'algorithme modifié appliqué à (F_{k+2}, F_{k+1}) donne naissance aux quotients 2, 3, 3, …, et aux restes modifiés F_{k-1}, F_{k-3}, F_{k-5}, … (avec $\varepsilon_1 = \varepsilon_2 = \cdots = -1$).