L'algorithme CORDIC

Daniel PERRIN

0. Introduction.

L'algorithme CORDIC (COordinate Rotation for DIgital Computer), mis au point en 1959 par l'américain Volder, permet de calculer, avec une précision fixée, les logarithmes népériens des nombres réels positifs (ni trop grands ni trop petits toutefois) à partir d'un nombre fini de valeurs supposées connues de la fonction logarithme. Il repose essentiellement sur l'équation fonctionnelle du logarithme : $\ln(xy) = \ln(x) + \ln(y)$.

1. Le principe.

On fixe une précision, disons 3×10^{-8}, (mais la méthode se généralise avec toute précision). On va calculer tous les nombres $\ln(x)$ pour x réel avec, disons, $10^{-1000} < x < 10^{1000}$ (au-delà, de toutes façons, les calculatrices usuelles considèrent que x est nul ou infini). Les nombres $\ln(x)$ seront calculés à partir de dix valeurs approchées $a \simeq \ln(10)$, $a_0 \simeq \ln(2)$, $a_1 \simeq \ln(1,1)$, $a_2 \simeq \ln(1,01)$, ..., $a_i \simeq \ln(1 + 10^{-i})$, ..., $a_8 \simeq \ln(1 + 10^{-8})$. **Attention**, ces valeurs sont supposées calculées au départ, par une autre méthode, série ou intégrale par exemple, et elles doivent être connues avec une précision supérieure à 10^{-8}, voir plus bas.

Le principe de la méthode est de se ramener au cas $1 < x < 10$ puis d'encadrer $10/x$:

$$y \leq \frac{10}{x} \leq y\,(1 + 10^{-8}),$$

avec un y de la forme

$$y = 2^{\alpha_0}(1,1)^{\alpha_1}(1,01)^{\alpha_2} \cdots (1 + 10^{-i})^{\alpha_i} \cdots (1 + 10^{-8})^{\alpha_8}$$

où les α_i sont des entiers ≥ 0, de taille bornée. On aura alors $\ln(x) \simeq \ln(10) - \ln(y)$ avec une erreur $\epsilon \leq \ln(1 + 10^{-8}) \leq 10^{-8}$.

Comme on connaît des valeurs approchées de $\ln(10)$ et de $\ln(y)$: $\ln(10) \simeq a$ et

$$\ln(y) = \alpha_0 \ln(2) + \alpha_1 \ln(1,1) + \cdots + \alpha_8 \ln(1 + 10^{-8}) \simeq \alpha_0 a_0 + \cdots + \alpha_8 a_8,$$

on en déduira une valeur approchée de $\ln(x)$ à 2×10^{-8} si les valeurs approchées a_i sont assez précises.

2. Une rédaction pour des élèves de terminale.

On commence par expliquer l'objectif de l'algorithme comme on l'a fait comme ci-dessus. On donne ensuite les a_i, valeurs approchées à 10^{-10} près : $\ln(10) \simeq a = 2,30258509299$, $\ln(2) \simeq a_0 = 0,6931471806$, $\ln(1,1) \simeq a_1 = 0,0953101798$, $\ln(1,01) \simeq a_2 = 0,0099503309$, $\ln(1,001) \simeq a_3 = 0,0009995003$, $\ln(1,0001) \simeq a_4 = 0,0000999950$, $\ln(1 + 10^{-5}) \simeq a_5 = 10^{-5}$, $\ln(1 + 10^{-6}) \simeq a_6 = 10^{-6}$, $\ln(1 + 10^{-7}) \simeq a_7 = 10^{-7}$ et $\ln(1 + 10^{-8}) \simeq a_8 = 10^{-8}$.

On peut calculer ces valeurs, par exemple par la méthode du point médian pour $\ln(10)$ *et* $\ln(2)$ *et par la série de* $\ln(1+x)$ *pour les autres. L'inégalité* $x - x^2/2 \leq \ln(1+x) \leq x$ *qui provient par exemple du développement en série explique les dernières valeurs.*

Voici maintenant la rédaction proprement dite de l'exercice :

1) Soit x un réel vérifiant $1 < x < 10$ (donc aussi $1 < \dfrac{10}{x} < 10$).

a) Montrer qu'il existe un entier α_0 avec $0 \leq \alpha_0 \leq 3$ tel que :

$$2^{\alpha_0} \leq \frac{10}{x} < 2^{\alpha_0+1}.$$

b) Montrer qu'il existe un entier α_1 avec $0 \leq \alpha_1 < 8$ tel que :

$$2^{\alpha_0}(1,1)^{\alpha_1} \leq \frac{10}{x} < 2^{\alpha_0}(1,1)^{\alpha_1+1}.$$

(On notera qu'on a $(1,1)^8 > 2$.)

c) Montrer par récurrence que, pour $i = 2, \cdots, 8$, il existe des entiers α_i avec $0 \leq \alpha_i < 10$ tels que :

$$2^{\alpha_0}(1,1)^{\alpha_1} \cdots (1+10^{-i})^{\alpha_i} \leq \frac{10}{x} < 2^{\alpha_0}(1,1)^{\alpha_1} \cdots (1+10^{-i})^{\alpha_i+1}.$$

(On notera, en développant par la formule du binôme, qu'on a $(1+10^{-i})^{10} > 1+10^{-(i-1)}$.)

d) On pose

$$y = 2^{\alpha_0}(1,1)^{\alpha_1}(1,01)^{\alpha_2} \cdots (1+10^{-i})^{\alpha_i} \cdots (1+10^{-8})^{\alpha_8}.$$

Montrer que

$$\ln(10) - \ln(y) = \ln(10) - \big[\alpha_0 \ln(2) + \alpha_1 \ln(1,1) + \cdots + \alpha_8 \ln(1+10^{-8})\big]$$

est une valeur approchée de $\ln(x)$ à 10^{-8} près (on rappelle qu'on a $\ln(1+x) \leq x$ pour tout $x \geq 0$). Les valeurs approchées a et a_i étant données avec une précision de 10^{-10}, montrer que $a - (\alpha_0 a_0 + \cdots + \alpha_8 a_8)$ est une valeur approchée de $\ln(x)$ à 2×10^{-8} près.

C'est ici qu'on voit l'importance du fait de borner les α_i. *On a une erreur de* 10^{-10} *sur chaque* a_i *donc, comme* α_i *est* < 10, *l'erreur sur* $\alpha_i a_i$ *est plus petite que* 10^{-9} *et comme il y a au plus* 10 *termes, l'erreur totale sur* $\ln(y)$ *est plus petite que* 10^{-8}.

e) Écrire un programme à la calculatrice implantant l'algorithme ci-dessus.
Donner une valeur approchée à 10^{-8} près de $\ln(4,4558911)$, de $\ln(5,63452)$.

Si l'on est paresseux, ou si l'on manque de temps, par exemple à l'oral du CAPES, on pourra, plutôt que de stocker les a_i, *tricher en demandant à la machine les valeurs des logarithmes correspondant (mais ceux-là seulement, évidemment.)*

2) Soit x un nombre réel vérifiant $10^{-1000} < x < 10^{1000}$. Montrer qu'il existe un entier n compris entre -1000 et 1000 tel que $1 < 10^{-n}x < 10$. En appliquant ce qui précède au calcul de $\ln(10^{-n}x)$, montrer qu'on peut calculer $\ln(x)$ avec une précision de 3×10^{-8} pourvu qu'on connaisse $\ln(10)$ à 10^{-11} près. Calculer $\ln(4567,8903)$.
Avec quelle précision faudrait-il connaître $\ln(10)$ pour avoir une valeur approchée à 10^{-8} près de $\ln(10^n)$ avec $n = 10^{10^{1000}}$?