Calcul de fonctions trigonométriques avec l'algorithme CORDIC et son implémentation

Pierre Langlois

CORDIC
Sujets de ce thème

- Origines
- Principe de base: rotation d’un vecteur
- Clé de la réalisation matérielle efficace
- Déroulement de l’algorithme
- Implémentation de l’algorithme et chemin des données

Calculer le sinus et le cosinus avec CORDIC

- CORDIC: COordinate Rotation DIgital Computer, proposé par Jack Volder, un ingénieur américain, en 1959.
- L’algorithme CORDIC permet de calculer les fonctions trigonométriques avec une suite d’opérations arithmétiques très simples: addition, soustraction et décalage.
- Utilisé dans les calculatrices de poche, Intel 80x87, 80486 et Motorola 68881.
- L’algorithme CORDIC a été généralisé pour calculer des fonctions exponentielles, la division, la multiplication et la racine carrée.

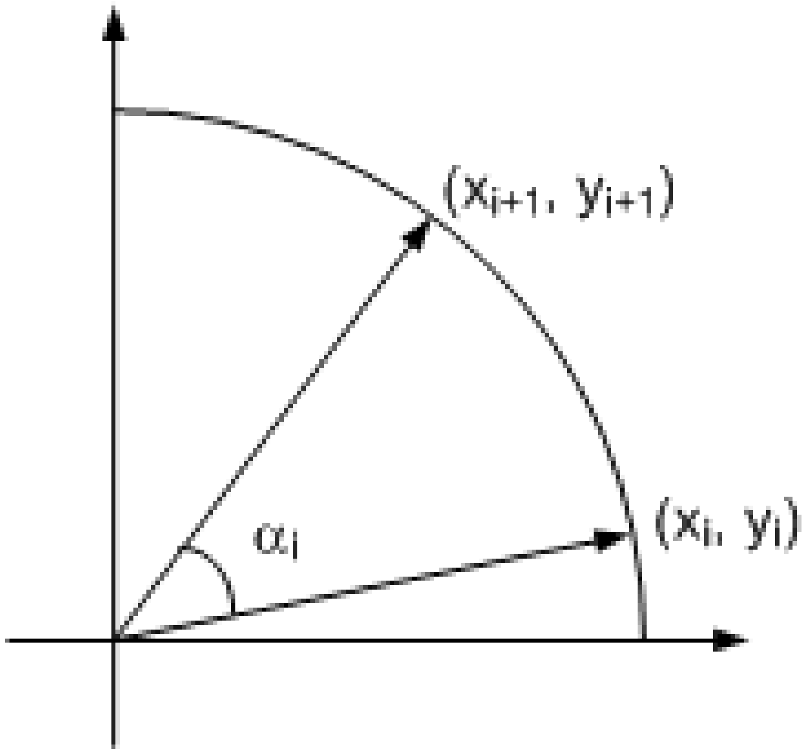
Principe de base

- Une rotation dans le plan peut s'effectuer à l'aide de l'équation suivante:

$$\begin{bmatrix} x_{i+1} \\ y_{i+1} \end{bmatrix} = \begin{bmatrix} \cos\alpha_i & -\sin\alpha_i \\ \sin\alpha_i & \cos\alpha_i \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix}$$

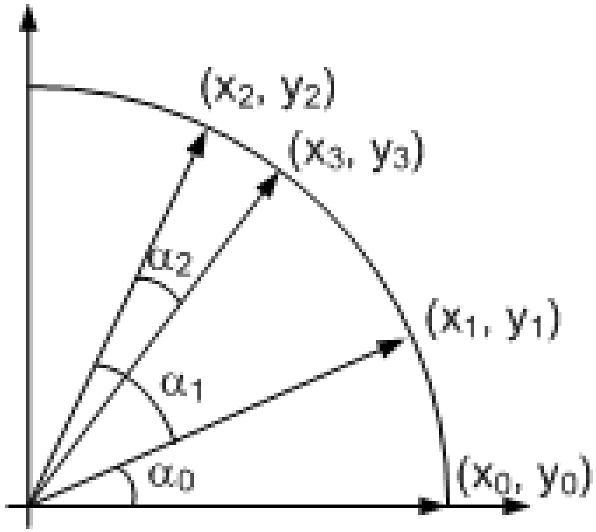
$$= \cos\alpha_i \begin{bmatrix} 1 & -\tan\alpha_i \\ \tan\alpha_i & 1 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix}$$

$$x_{i+1} = \cos\alpha_i (x_i - y_i \tan\alpha_i)$$
$$y_{i+1} = \cos\alpha_i (y_i + x_i \tan\alpha_i)$$



Principe de base

- On peut décomposer une rotation en plusieurs sous-rotations.
  - L'angle global de rotation est égal à la somme des angles des sous-rotations.
  - Les angles des sous-rotations peuvent être positifs ou négatifs.



$$\begin{bmatrix} x_3 \\ y_3 \end{bmatrix} = \cos\alpha_2 \cos\alpha_1 \cos\alpha_0 \times$$

$$\begin{bmatrix} 1 & -\tan\alpha_2 \\ \tan\alpha_2 & 1 \end{bmatrix} \begin{bmatrix} 1 & -\tan\alpha_1 \\ \tan\alpha_1 & 1 \end{bmatrix} \begin{bmatrix} 1 & -\tan\alpha_0 \\ \tan\alpha_0 & 1 \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \end{bmatrix}$$

$$\begin{bmatrix} x_n \\ y_n \end{bmatrix} = K \times \begin{bmatrix} 1 & -\tan\alpha_{n-1} \\ \tan\alpha_{n-1} & 1 \end{bmatrix} \begin{bmatrix} 1 & -\tan\alpha_{n-2} \\ \tan\alpha_{n-2} & 1 \end{bmatrix} \cdots \begin{bmatrix} 1 & -\tan\alpha_0 \\ \tan\alpha_0 & 1 \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \end{bmatrix}$$

$$K = \prod_{0}^{n-1} \cos\alpha_i$$

Clé de la réalisation matérielle efficace

- **La clé de la réalisation matérielle simple de l'algorithme CORDIC est qu'on choisit tan(α_i) = 2^{-i}.**
- Dans l'implémentation des équations, la multiplication par tan(α_i) peut donc être effectuée par un simple décalage de bits.
- Comme on choisit les α_i, on peut calculer d'avance les cos α_i ainsi que *K*, le produit des cos α_i.

i	tan(α_i)	α_i (degrés)
0	1.000	45.0000
1	0.500	26.5651
2	0.250	14.0362
3	0.125	7.1250

$$\begin{bmatrix} x_n \\ y_n \end{bmatrix} = K \times \begin{bmatrix} 1 & -\tan\alpha_{n-1} \\ \tan\alpha_{n-1} & 1 \end{bmatrix} \begin{bmatrix} 1 & -\tan\alpha_{n-2} \\ \tan\alpha_{n-2} & 1 \end{bmatrix} \cdots \begin{bmatrix} 1 & -\tan\alpha_0 \\ \tan\alpha_0 & 1 \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \end{bmatrix}$$

$$K = \prod_{0}^{n-1} \cos\alpha_i$$

Déroulement de l'algorithme

- Pour obtenir le sinus et le cosinus d'un angle, on prend le vecteur (1, 0) comme point de départ et on le fait tourner par l'angle désiré.
- Les coordonnées (*x*, *y*) obtenues sont le cosinus et le sinus de l'angle, respectivement.
- Dans l'algorithme CORDIC, on prend le vecteur (*K*, 0) comme point de départ.
- Il reste à trouver la somme des angles α_i qui est égale à l'angle désiré *z*.

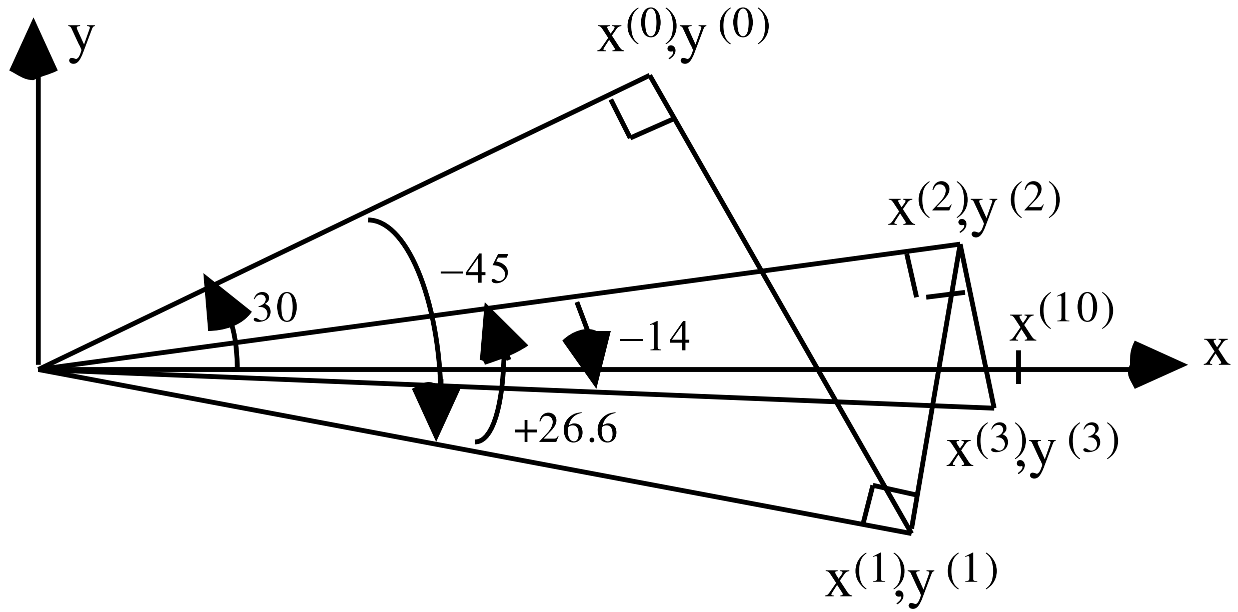
$$z \cong \sum_{0}^{n-1} d_i \alpha_i, d_i \in \{-1, 1\}$$

$$\begin{bmatrix} \cos z \\ \sin z \end{bmatrix} \cong \begin{bmatrix} 1 & -d_{n-1} \tan \alpha_{n-1} \\ d_{n-1} \tan \alpha_{n-1} & 1 \end{bmatrix} \begin{bmatrix} 1 & -d_{n-2} \tan \alpha_{n-2} \\ d_{n-2} \tan \alpha_{n-2} & 1 \end{bmatrix} \cdots$$

$$\cdots \begin{bmatrix} 1 & -d_0 \tan \alpha_0 \\ d_0 \tan \alpha_0 & 1 \end{bmatrix} \begin{bmatrix} K \\ 0 \end{bmatrix}$$

Déroulement de l'algorithme

- En pratique, on procède à l'envers: on part de l'angle z et on fait des rotations par les angles α_i (positives ou négatives) jusqu'à ce qu'on arrive à 0.
- On calcule $z_{i+1} = z_i - d_i \alpha_i$
- Le signe de la rotation d_i est égal au signe de l'angle courant z_i.
- Exemple des trois premières rotations pour l'angle *z* = 30 degrés.



B. Parhami, Computer Arithmetic, Oxford University Press, 2000.

Déroulement de l'algorithme

- On obtient finalement trois équations à implémenter.
- Les opérations requises sont l'addition/soustraction et le décalage.
- Un tableau doit contenir les α_i, mais on note que pour α_i petit, $\tan \alpha_i = \alpha_i = 2^{-i}$ par choix.
- Il faut déterminer le nombre d'itérations à faire, on obtient environ un bit de précision par itération.

$$x_{i+1} = (x_i - d_i y_i \tan \alpha_i)$$

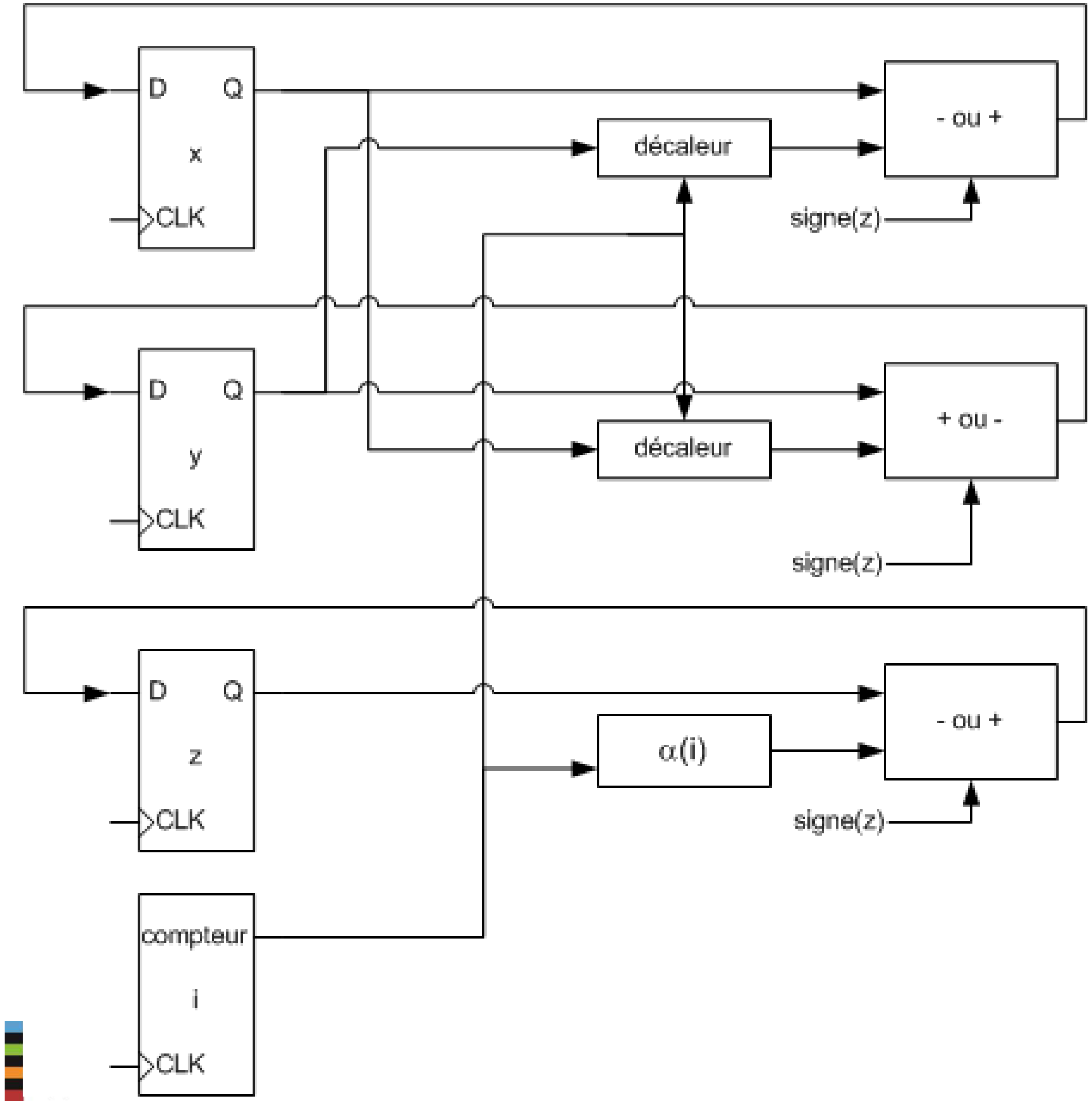
$$y_{i+1} = (y_i + d_i x_i \tan \alpha_i)$$

$$z_{i+1} = z_i - d_i \alpha_i$$

Analyse du problème pour l'implémentation

- Les ports du circuit de transmission sont:
  - reset, clk
  - theta_rad (entrée): l'angle z exprimé en radians, limité entre $-\pi/4$ et $\pi/4$.
  - go(entrée): indique que l'angle dont on veut obtenir le sinus et le cosinus est placé sur le port theta_rad et que les calculs peuvent débuter
  - pret (sortie): indique que les calculs sont terminés
  - costheta et sintheta (sorties): les résultats
- Toutes les valeurs sont fractionnaires.
- Besoin de cinq éléments à mémoire:
  - trois registres pour x, y et z
  - un registre d'états:
    - en train de faire les calculs: pret <= '0' et on n'accepte pas de nouvel angle
    - en attente: pret <= '1' et on accepte un nouvel angle
  - un compteur interne pour déterminer si on a fait toutes les itérations

CORDIC
Chemin des données



$$x_{i+1} = (x_i - d_i y_i \tan \alpha_i)$$

$$y_{i+1} = (y_i + d_i x_i \tan \alpha_i)$$

$$z_{i+1} = z_i - d_i \alpha_i$$

$$d_i = \mathrm{sgn}(z_i)$$

Vous devriez maintenant être capable de …

- Expliquer les principes de l'algorithme CORDIC. (B2)
- Calculer le sinus et le cosinus d'un nombre à l'aide de l'algorithme CORDIC. (B3)

Code	Niveau (http://fr.wikipedia.org/wiki/Taxonomie_de_Bloom)
B1	Connaissance - mémoriser de l'information.
B2	Compréhension – interpréter l'information.
B3	Application – confronter les connaissances à des cas pratiques simples.
B4	Analyse – décomposer un problème, cas pratiques plus complexes.
B5	Synthèse – expression personnelle, cas pratiques plus complexes.