Project 1 - Approximations of Sine and Cosine Using the CORDIC Algorithm

How do calculators compute sines and cosines? Naturally, you will think to use the power series representations of sine and cosine

\[
\sin(x) = x - \frac{1}{3!}x^3 + \frac{1}{5!}x^5 - \cdots = \sum_{k=1}^{\infty} (-1)^{k-1} \frac{1}{(2k-1)!}x^{2k-1}
\]

\[
\cos(x) = 1 - \frac{1}{2!}x^2 + \frac{1}{4!}x^4 - \cdots = \sum_{k=0}^{\infty} (-1)^{k+1} \frac{1}{(2k)!}x^{2k}.
\]

that you have learned in Calculus II. For example, the series for \(\sin(1)\) and \(\cos(1)\) are (let \(x = 1\))

\[
\sin(1) = 1 - \frac{1}{3!} + \frac{1}{5!} - \cdots = \sum_{k=1}^{\infty} (-1)^{k-1} \frac{1}{(2k-1)!}
\]

\[
\cos(1) = 1 - \frac{1}{2!} + \frac{1}{4!} - \cdots = \sum_{k=0}^{\infty} (-1)^{k+1} \frac{1}{(2k)!}.
\]

For a given accuracy requirement \(\varepsilon\), we find \(N_1\) and \(N_2\) as discussed in the last section, which are the smallest positive integers such that

\[
b_{N_1+1} = \frac{1}{(2N_1 + 1)!} < \varepsilon \quad \text{and} \quad b_{N_2+1} = \frac{1}{(2N_2 + 1)!} < \varepsilon.
\]

Then

\[
\sin(1) \approx \sum_{k=1}^{N_1} (-1)^{k-1} \frac{1}{(2k-1)!} \quad \text{and} \quad \cos(1) \approx \sum_{k=1}^{N_2} (-1)^{k-1} \frac{1}{(2k-2)!}.
\]

Actually, calculators do not use partial sums of series to approximate sines and cosines. The algorithm implemented in calculators for sines and cosines is called CORDIC (COordinate Rotation Digital Computer). CORDIC was first discovered by Jack Volder in 1959 and was developed to replace the analog resolver in the B-58 bomber’s navigation computer. To read more about the history of this algorithm, please check out the following two references:

(1) http://en.wikipedia.org/wiki/CORDIC

In this lecture, we study in detail how CORDIC approximates \(\sin(\theta)\) and \(\cos(\theta)\).

For a given \(-\frac{\pi}{2} < \theta \leq \frac{\pi}{2}\) in radians, the algorithm CORDIC takes two steps to compute the approximations of \(\sin(\theta)\) and \(\cos(\theta)\). Let \(\theta_i = \tan^{-1}\left(\frac{1}{2i-1}\right)\) for \(i = 1, 2, \ldots, 40\).

**Step 1.** Approximate \(\theta\) by \(\sum_{i=1}^{40} \sigma_i \theta_i\) where

\[
\sigma_i = 1 \text{ if } \theta - \sum_{j=1}^{i-1} \sigma_j \theta_j > 0 \quad \text{and} \quad \sigma_i = -1 \text{ if } \theta - \sum_{j=1}^{i-1} \sigma_j \theta_j < 0
\]

\[\sigma_1 = 1 \text{ if } \theta > 0 \text{ and } \sigma_1 = -1 \text{ if } \theta < 0.\]

**Step 2.** \(\cos(\theta) \approx \cos\left(\sum_{i=1}^{40} \sigma_i \theta_i\right)\) and \(\sin(\theta) \approx \sin\left(\sum_{i=1}^{40} \sigma_i \theta_i\right)\).

The computations given in Step 2. seem more complicated than the ones for \(\cos(\theta)\) and \(\sin(\theta)\). Actually, both \(\cos\left(\sum_{i=1}^{40} \sigma_i \theta_i\right)\) and \(\sin\left(\sum_{i=1}^{40} \sigma_i \theta_i\right)\) can be computed iteratively, and can also be simplified using several trigonometric identities and the definitions \(\theta_i = \tan^{-1}\left(\frac{1}{2i-1}\right)\) for \(i = 1, 2, \ldots, 40\). By the sum and difference formulas for sine and cosine
\[
\sin(\alpha \pm \beta) = \sin(\alpha) \cos(\beta) \pm \sin(\beta) \cos(\alpha)
\]
\[
\cos(\alpha \pm \beta) = \cos(\alpha) \cos(\beta) \mp \sin(\alpha) \sin(\beta),
\]

we have for \( k = 2, \ldots, 40, \)
\[
\sin\left(\sum_{i=1}^{k} \sigma_i \theta_i\right) = \sin\left(\sigma_k \theta_k + \sum_{i=1}^{k-1} \sigma_i \theta_i\right) = \sin(\sigma_k \theta_k) \cos\left(\sum_{i=1}^{k-1} \sigma_i \theta_i\right) + \sin\left(\sum_{i=1}^{k-1} \sigma_i \theta_i\right) \cos(\sigma_k \theta_k)
\]
\[
\cos\left(\sum_{i=1}^{k} \sigma_i \theta_i\right) = \cos\left(\sigma_k \theta_k + \sum_{i=1}^{k-1} \sigma_i \theta_i\right) = \cos(\sigma_k \theta_k) \cos\left(\sum_{i=1}^{k-1} \sigma_i \theta_i\right) - \sin\left(\sum_{i=1}^{k-1} \sigma_i \theta_i\right) \sin(\sigma_k \theta_k).
\]

Let \( s_0 = 1 \) and \( c_0 = 0 \). Define that \( s_k = \sin\left(\sum_{i=1}^{k} \sigma_i \theta_i\right) \) and \( c_k = \cos\left(\sum_{i=1}^{k} \sigma_i \theta_i\right) \). Then for \( k = 1, \ldots, 40, \)
we have the following recurrence relations:
\[
c_k = \cos(\sigma_k \theta_k) c_{k-1} - \sin(\sigma_k \theta_k) s_{k-1} \quad \text{and} \quad s_k = \sin(\sigma_k \theta_k) c_{k-1} + \cos(\sigma_k \theta_k) s_{k-1}.
\]
Since \( \sigma_k = 1 \) or \(-1, \) and \( \sin(-\alpha) = -\sin(\alpha) \) and \( \cos(-\alpha) = \cos(\alpha), \)
\[
\sin(\sigma_k \theta_k) = \sigma_k \sin(\theta_k) \quad \text{and} \quad \cos(\sigma_k \theta_k) = \cos(\theta_k).
\]
The above recurrence relations become
\[
c_k = \cos(\theta_k) c_{k-1} - \sigma_k \sin(\theta_k) s_{k-1} \quad \text{and} \quad s_k = \sigma_k \sin(\theta_k) c_{k-1} + \cos(\theta_k) s_{k-1}.
\]
Hence, \( c_{40} \) and \( s_{40} \) can be computed iteratively.

Can we simplify the computations of \( c_k \) and \( s_k \)? Observe that
\[
c_k = \cos(\theta_k) \left( c_{k-1} - \sigma_k \frac{\sin(\theta_k)}{\cos(\theta_k)} s_{k-1} \right) = \cos(\theta_k) (c_{k-1} - \sigma_k \tan(\theta_k) s_{k-1}) \quad \text{and} \quad s_k = \cos(\theta_k) \left( \sigma_k \frac{\sin(\theta_k)}{\cos(\theta_k)} c_{k-1} + s_{k-1} \right) = \cos(\theta_k) (\sigma_k \tan(\theta_k) c_{k-1} + s_{k-1}).
\]

Because of the trigonometric identities: (i) \( \cos(\alpha) = \frac{1}{\sec(\alpha)} \) and (ii) \( \sec^2(\alpha) - \tan^2(\alpha) = 1, \)
we have
\[
\cos(\theta_k) = \frac{1}{\sec(\theta_k)} = \frac{1}{\sqrt{1 + \tan^2(\theta_k)}}.
\]
Since \( \theta_i = \tan^{-1}\left(\frac{1}{2^{i-1}}\right), \) \( \tan(\theta_i) = \frac{1}{2^{i-1}} = 2^{-(i-1)} \) and \( \cos(\theta_i) = \frac{1}{\sqrt{1 + \left(\frac{1}{2^{i-1}}\right)^2}} = \frac{1}{\sqrt{1 + 2^{-2(i-1)}}}. \)

Substituting these identities into the recurrence relations of \( c_k \) and \( s_k, \) we have
\[
c_k = \frac{1}{\sqrt{1 + 2^{-2(k-1)}}} (c_{k-1} - \sigma_k 2^{-(k-1)} s_{k-1}) \quad \text{and} \quad s_k = \frac{1}{\sqrt{1 + 2^{-2(k-1)}}} (\sigma_k 2^{-(k-1)} c_{k-1} + s_{k-1}).
\]
Can we further simplify \( c_k \) and \( s_k? \) Yes. First we express the recurrence relations of \( c_k \) and \( s_k \) in matrix and vector notations.
\[
\begin{bmatrix} c_k \\ s_k \end{bmatrix} = \frac{1}{\sqrt{1 + 2^{-2(k-1)}}} \begin{bmatrix} 1 & -\sigma_k 2^{-(k-1)} \\ \sigma_k 2^{-(k-1)} & 1 \end{bmatrix} \begin{bmatrix} c_{k-1} \\ s_{k-1} \end{bmatrix}.
\]
Observe that
\[
\begin{bmatrix}
  c_k \\
  s_k
\end{bmatrix} = \frac{1}{\sqrt{1 + 2^{-2(k-1)}}} \begin{bmatrix}
  1 & -\sigma_k 2^{-(k-1)} \\
  \sigma_k 2^{-(k-1)} & 1
\end{bmatrix} \begin{bmatrix}
  c_{k-1} \\
  s_{k-1}
\end{bmatrix}
\]
\[
= \frac{1}{\sqrt{1 + 2^{-2(k-1)}}} \begin{bmatrix}
  1 & -\sigma_k 2^{-(k-1)} \\
  \sigma_k 2^{-(k-1)} & 1
\end{bmatrix} \left( \frac{1}{\sqrt{1 + 2^{-2(k-2)}}} \begin{bmatrix}
  1 & -\sigma_{k-1} 2^{-(k-2)} \\
  \sigma_{k-1} 2^{-(k-2)} & 1
\end{bmatrix} \right) \begin{bmatrix}
  c_{k-2} \\
  s_{k-2}
\end{bmatrix}
\]
\[
\vdots
\]
\[
= \frac{1}{\sqrt{1 + 2^{-2(k-1)}}} \cdots \frac{1}{\sqrt{1 + 2^{-2(0)}}} \begin{bmatrix}
  1 & -\sigma_k 2^{-(k-1)} \\
  \sigma_k 2^{-(k-1)} & 1
\end{bmatrix} \cdots \begin{bmatrix}
  1 & -\sigma_0 2^{-(0)} \\
  \sigma_0 2^{-(0)} & 1
\end{bmatrix} \begin{bmatrix}
  c_0 \\
  s_0
\end{bmatrix}.
\]

Let \( K = \frac{1}{\sqrt{1 + 2^{-2(39)}}} \cdots \frac{1}{\sqrt{1 + 2^{-2(0)}}} = \prod_{k=1}^{40} \frac{1}{\sqrt{1 + 2^{-2(k-1)}}} \). Note that \( K \) is an irrational number and \( K \approx 0.60725293500888126\ldots \).

Since calculators are in finite digits format, \( K \) can be implemented in a calculator according to its accuracy. For example, IT-89 has 15-digit accuracy so \( K \approx 0.607252935008881 \) is implemented to approximate sines and cosines. The second step of CORDIC is finally simplified as follows.

**Step 2.** Let \( \begin{bmatrix}
  \hat{c}_0 \\
  \hat{s}_0
\end{bmatrix} = \begin{bmatrix}
  1 \\
  0
\end{bmatrix} \). For \( k = 1, 2, \ldots, 40 \),
\[
\begin{bmatrix}
  \hat{c}_k \\
  \hat{s}_k
\end{bmatrix} = \begin{bmatrix}
  1 & -\sigma_k 2^{-(k-1)} \\
  \sigma_k 2^{-(k-1)} & 1
\end{bmatrix} \begin{bmatrix}
  \hat{c}_{k-1} \\
  \hat{s}_{k-1}
\end{bmatrix}.
\]
\[
\begin{bmatrix}
  c_{40} \\
  s_{40}
\end{bmatrix} = 0.607252935008881 \begin{bmatrix}
  \hat{c}_{40} \\
  \hat{s}_{40}
\end{bmatrix}.
\]

\( \cos(\theta) \approx c_{40} \) and \( \sin(\theta) \approx s_{40} \).

We implement the algorithm of CORDIC in MatLab and call the M-file cordic.m:
To run the program to approximate \( \sin(1), \cos(1), \sin(-0.1) \) and \( \cos(-0.1) \):
The algorithm of CORDIC is designed only for $-\frac{\pi}{2} < \theta \leq \frac{\pi}{2}$. In order to evaluate sines and cosines for all angles, we need to add a preliminary step to determine an angle $\bar{\theta}$ in $(-\frac{\pi}{2}, \frac{\pi}{2}]$ such that 
\[
\cos(\theta) = \cos(\bar{\theta}) \quad \text{or} \quad \cos(\theta) = -\cos(\bar{\theta}) \quad \text{and} \quad \sin(\theta) = \sin(\bar{\theta}) \quad \text{or} \quad \sin(\theta) = -\sin(\bar{\theta})
\]
depending on the quadrant containing the input angle. For a given $\theta$ in radians, there exist integer $m$ and the angle $0 \leq \bar{\theta} < 2\pi$ such that
\[
\theta = 2m\pi + \bar{\theta}.
\]

(i) If $0 \leq \bar{\theta} < \frac{\pi}{2}$ (the first quadrant), then let $\bar{\theta} = \hat{\theta}$ and $\sin(\theta) \approx \sin(\bar{\theta})$ and $\cos(\theta) \approx \cos(\bar{\theta})$.

(ii) If $\frac{\pi}{2} \leq \bar{\theta} < \pi$ (the second quadrant), then let $\bar{\theta} = \pi - \hat{\theta}$ and $\sin(\theta) \approx \sin(\bar{\theta})$ and $\cos(\theta) \approx -\cos(\bar{\theta})$.

(iii) If $\pi \leq \bar{\theta} < \frac{3\pi}{2}$ (the third quadrant), then let $\bar{\theta} = \hat{\theta} - \pi$ and $\sin(\theta) \approx -\sin(\bar{\theta})$ and $\cos(\theta) \approx -\cos(\bar{\theta})$.

(iv) If $\frac{3\pi}{2} \leq \bar{\theta} < 2\pi$ (the 4th quadrant), then let $\bar{\theta} = 2\pi - \hat{\theta}$ and $\sin(\theta) \approx -\sin(\bar{\theta})$ and $\cos(\theta) \approx \cos(\bar{\theta})$.

For example, let $\theta = 2$ radians which is in the second quadrant so $\bar{\theta} = \hat{\theta}$ and $\bar{\theta} = \pi - 2 = 1.14159265$.

For example, let $\theta = 10$ radians which is greater than $2\pi$. Then $\bar{\theta} = 10 - 2\pi = 3.71681469$ which is in the third quadrant so $\bar{\theta} = 3.71681469 - \pi = 0.57522203$.

We will end this section with a few important notes.
CORDIC approximates \( \sin(\theta) \) and \( \cos(\theta) \). Other four trigonometric functions can be approximated by the trigonometric identities:

\[
\tan(\theta) = \frac{\sin(\theta)}{\cos(\theta)}, \quad \cot(\theta) = \frac{\cos(\theta)}{\sin(\theta)}, \quad \sec(\theta) = \frac{1}{\cos(\theta)} \quad \text{and} \quad \csc(\theta) = \frac{1}{\sin(\theta)}.
\]

As stated at the beginning of the lecture, CORDIC stands for COordinate Rotation Digital Computer. How does the algorithm relate to rotations? Using the matrix and vector notations for the first version of the recurrence relations of \( c_k \) and \( s_k \):

\[
c_k = \cos(\sigma_k \theta_k) c_{k-1} - \sin(\sigma_k \theta_k) s_{k-1} \quad \text{and} \quad s_k = \sin(\sigma_k \theta_k) c_{k-1} + \cos(\sigma_k \theta_k) s_{k-1},
\]

we have

\[
\begin{bmatrix}
  c_k \\
  s_k
\end{bmatrix} =
\begin{bmatrix}
  \cos(\sigma_k \theta_k) & -\sin(\sigma_k \theta_k) \\
  \sin(\sigma_k \theta_k) & \cos(\sigma_k \theta_k)
\end{bmatrix}
\begin{bmatrix}
  c_{k-1} \\
  s_{k-1}
\end{bmatrix} = R_k
\begin{bmatrix}
  c_{k-1} \\
  s_{k-1}
\end{bmatrix}.
\]

Notice that the matrix \( R_k \) is a rotation matrix that rotates the vector \( \begin{bmatrix} c_{k-1} \\ s_{k-1} \end{bmatrix} \) by \( \theta_k \) counterclockwise if \( \sigma_k \theta_k \geq 0 \) or clockwise if \( \sigma_k \theta_k < 0 \). The CORDIC algorithm can also be viewed as an algorithm that approximates sines and cosines by rotating the vector \( \begin{bmatrix} 1 \\ 0 \end{bmatrix} \) to \( \begin{bmatrix} \cos(\theta_{40}) \\ \sin(\theta_{40}) \end{bmatrix} \) through angles \( \theta_1, \theta_2, \ldots, \theta_{40} \). We will study the efficiency of CORDIC later using the Fixed-point Algorithm.

**Exercise:**

1. How many angles in \( \left( -\frac{\pi}{2}, \frac{\pi}{2} \right) \) can be represented exactly by the form \( \sum_{k=1}^{40} \sigma_k \theta_k \) where \( \sigma_k \) is 1 or \(-1\)?

2. Suppose that we approximate a given \( \theta \) by only \( \theta_0, \theta_1, \theta_2, \theta_3, \) and \( \theta_4 \). Using CORDIC algorithm to approximate \( \theta = 0.6 \) and \( \theta = -1.1 \).

3. Use the MatLab M-file cordic.m to calculate \( \theta = 15 \) radians and \( \theta = -10 \) radians. Note that both angles are not in the interval \( \left( -\frac{\pi}{2}, \frac{\pi}{2} \right) \). Find \( \bar{\theta} \) and \( \bar{\theta} \) first and report both \( \cos(\theta) \) and \( \sin(\theta) \) using cordic.m. To make a copy of the M-file cordic.m, please click on it, copy/paste the content and save it in your current directory.