ALPHA3
HP49g Series library

Vincent Blay Roger

The library ALPHA3 for HP 49g series calculators computes a series of **parameters** needed to solve the **unidirectional molecular transport equations under transient state**.

The equation solved to compute these parameters, $F(\mathrm{X})$, depends on the geometry of the system. For rectangular or spherical coordinates, the equation is as follows:

$$F(X) = \begin{cases} \arctan\left(\frac{1}{m\cdot X}\right) - X + (i-1)\cdot\pi \;\; if \;\; C=1 \\ \\ \arctan\left(\frac{m\cdot X}{m-1}\right. \;\; - X + (i-c)\cdot\pi \;\; if \;\; C=0 \end{cases}$$

where: m is the dimensionless external transport resistance ($m \in \mathbb{R}^{+}$)
$C=1$ for rectangular coordinates or C = 0 for spherical coordinates
$c=1$ if $m>1$ or c = 0 otherwise
$i=1\ldots n$, where n *is* the number of parameters to be computed

If the dimensionless external resistance can be neglected (m = 0), or if m = 1 in spherical coordinates, the solution is trivial. The list of parameters is then readily generated with the built-in function xSEQ:

$$X = \begin{cases} (i - C/2) & if \;\; m=0 \\ (i-1/2)\cdot\pi & if\, C=0 \text{ and } m=1 \end{cases}$$

In the general case, however, the equation F(X) = 0 is solved with the built-in command xROOT, which uses the bisection method.

In the case of cylindrical coordinates, the function $F(\mathrm{X})$ involves the Bessel functions $J_1(X)$ and $J_0(X)$:

$$F(X) = X\frac{J_1(X)}{J_0(X)} - \frac{1}{m} \tag{1}$$

The ROM of the HP 49g series calculators does not include routines to compute the Bessel functions. In the library ALPHA3, we estimate the Bessel functions as truncated series expansions. Both J_1 and J_0 at a given point X are computed in a same loop, as shown in the scheme and SysRPL code below.

Notice that the absolute tolerance for the estimate of the Bessel function J_0 is $\pm 1 \times 10^{-8}$, which is set to be lower than the tolerance required to the numerical method to solve $F(\mathrm{X}) = 0$ itself ($\pm 1 \times 10^{-6}$).

$$
\begin{array}{llllllll}
\mathrm{J}_0(X) = & 1 & & -\dfrac{(X/2)^2}{1!1!} & & +\dfrac{(X/2)^4}{2!2!} & & +\ldots \\
& & {\scriptstyle \times(X/2)\div 1} \searrow \quad {\scriptstyle \times(-X/2)\div 1} \nearrow & & {\scriptstyle \times(X/2)\div 2} \searrow \quad {\scriptstyle \times(-X/2)\div 2} \nearrow & & {\scriptstyle \times(X/2)\div 3} \searrow & \\
\mathrm{J}_1(X) = & & \dfrac{(X/2)}{1!} & & -\dfrac{(X/2)^3}{1!2!} & & +\dfrac{(X/2)^5}{2!3!} & +\ldots
\end{array}
$$

The code section is as follows:

```
NULLNAME JN
::
LAM X %2 %/ %1 ZEROZEROTWO DOBIND
%1
BEGIN
2GETLAM %* 1GETLAM %/
DUP LAM J1 %+ ' LAM J1 STO
2GETLAM %CHS %* 1GETLAM %/
DUPDUP LAM J0 %+ ' LAM J0 STO
%ABS 1E-8 %<
1GETLAM %1+ 1PUTLAM
UNTIL
DROP
ABND
;
```

To find the zeros of F(X) in cylindrical coordinates, the library ALPHA3 employs the Newton-Raphson method. To compute the derivative of the function F(X) it makes use of the identities:

$$\frac{d}{dX}\mathrm{J}_0(X) = -\mathrm{J}_1(X)$$

$$\frac{d}{dX}\mathrm{J}_\mathrm{p}(\alpha X) = \alpha \mathrm{J}_{\mathrm{p}-1}(\alpha X) - \frac{p}{X}\mathrm{J}_\mathrm{p}(\alpha X)$$

which leads to the following result:

$$F(X) = m \cdot X \cdot \mathrm{J}_1(X) - \mathrm{J}_0(X)$$

$$F'(X) = m \cdot X \cdot \mathrm{J}_0(X) + \mathrm{J}_1(X)$$

This is the preferred form to write the function $F(X)$, as it improves the convergence of the numeric method to find the roots of the function. This can be observed in the following plots (for $m = 1$). As the starting guess for the first zero of F, we use a value of X= π/2. For the subsequent zeros, we use the value of the last root computed + π.

The starting function $F(\mathrm{X})$ defined by (1) has the following shape:

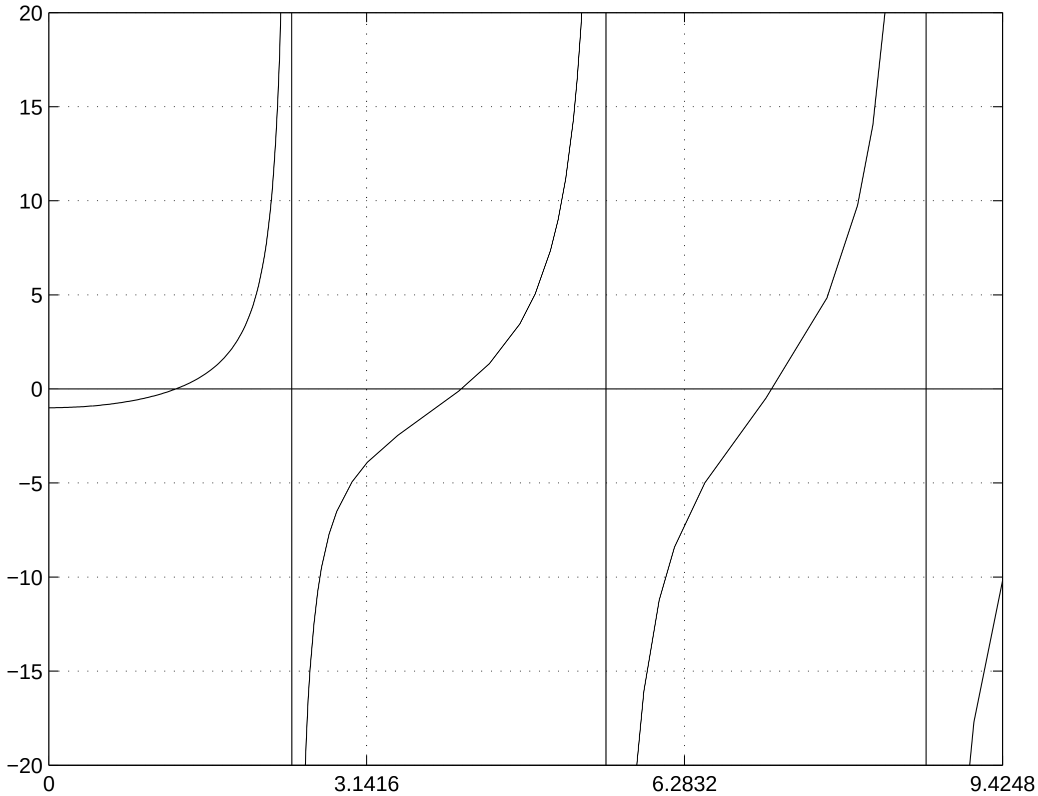

Figure 1: Plot of function (1)

Another possibility to re-write the function could be:

$$F(X) = \frac{\mathrm{J}_0(X)}{\mathrm{J}_1(X)} - m \cdot X \tag{2}$$

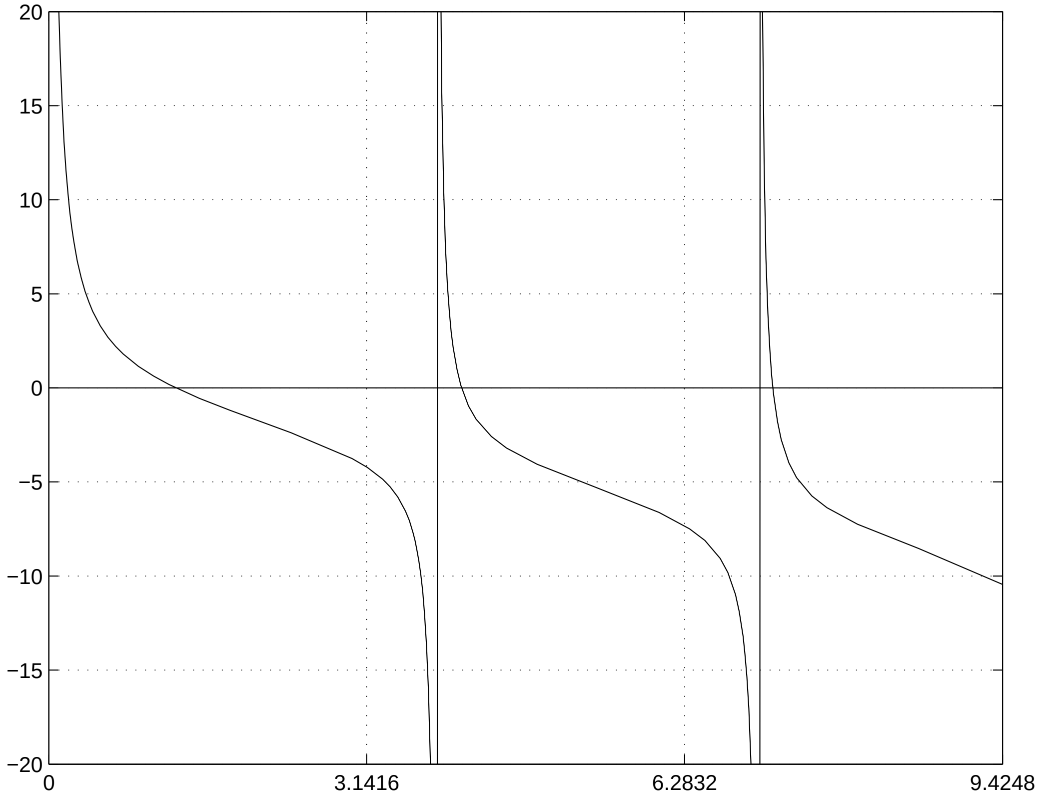

Figure 2: Plot of function (2)

It can be seen that the slope of both forms (1) and (2) varies widely. The changes in the convexity of the function can cause the Newton method to converge to roots far away from the initial guess or seed value, to return the same root for several iterations, or even to return negative numbers (remember that the Bessel functions are estimated given positive real arguments in this library by truncating a series expansion). The vertical jumps correspond to discontinuities in the function.

Re-writing F(X) as proposed next, a much softer behavior is obtained, which is more appropriate for the Netwon method to solve $F(\mathrm{X}) = 0$:

$$F(X) = m \cdot X \cdot \mathrm{J}_1(X) - \mathrm{J}_0(X) \tag{3}$$

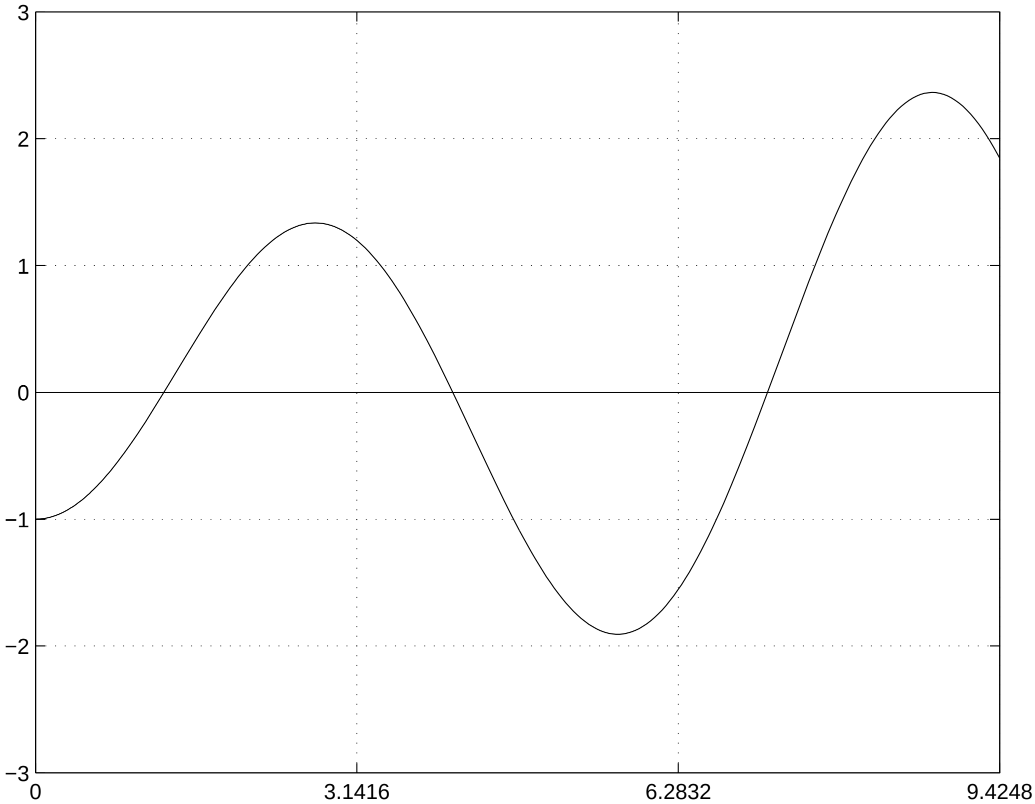

Figure 3: Plot of function (3)