A Tan Root for Battin's Lambert Recipe

Randal Allen[1] and Russel Wenzel[2]

I.Background

RICHARD Battin's approach to the solution of the Lambert Problem requires evaluation of a hypergeometric series expressed as an infinite continued fraction (Battin, p. 328, eq. 7.96):

$$\xi(x) = 5 + \cfrac{9x}{7 + \cfrac{16x}{9 + \cfrac{25x}{11 + \cfrac{36x}{13 + \ddots}}}} \qquad x \in (-1, \infty) \tag{1}$$

Typically, approximate evaluations of infinite continued fractions are made by encoding algorithms which either evaluate a pre-determined (finite) number of terms, or convergents, from the "Bottom Up" (Battin, p. 64) or involve an iterative approach which monitors the ratio of successive convergents until it falls within a specified tolerance, i.e. "Top Down" (Battin, p. 68). Both have their advantages and disadvantages, but perhaps the biggest challenge to a developer seeking a quick means of evaluating Battin's series is that either approach requires him to construct a specialized evaluation subroutine. Such routines can at best be costly in terms of execution time, or at worst, costly to learn, implement and possibly still return inaccurate results.

As such, we consider here an alternate and arguably a more practical method of evaluating Battin's series. Here, "practical" is defined with regards to terms that it doesn't require the user to know much about the more subtler aspects of the "convergence of the convergents", if you will. Instead, it takes advantage of functions already typically intrinsic to most modern computer languages; series implementations already encoded at the machine level, optimized to guarantee convergence for a broad range of arguments, and perhaps best of all: tested and proven for years.

[1] Lone Star, Addison, TX, AIAA Associate Fellow
[2] Raytheon, Aurora, CO

We first consider another continued fraction which Battin related to his series defined in equation (1) (Battin, p. 327, eq. 7.93):

$$F\left(\tfrac{1}{2},1;\tfrac{3}{2};-x\right)=\cfrac{1}{1+\cfrac{x}{3+\cfrac{4x}{5+\cfrac{9x}{7+\cfrac{16x}{9+\ddots}}}}} \tag{2}$$

Battin also shows that (Battin, p. 49, Problem 1-10):

$$\tan^{-1}(x)=\cfrac{x}{1+\cfrac{x^2}{3+\cfrac{(2x)^2}{5+\cfrac{(3x)^2}{7+\ddots}}}} \qquad \tanh^{-1}(x)=\cfrac{x}{1-\cfrac{x^2}{3-\cfrac{(2x)^2}{5-\cfrac{(3x)^2}{7-\ddots}}}} \tag{3}$$

It isn't too difficult to see that we can combine the equations in (3) and relate them to equation (2) by a domain partition in x, in the form we designate

$$F\left(\tfrac{1}{2},1;\tfrac{3}{2};-x\right)=ATANR(x)=\begin{cases}\dfrac{\tanh^{-1}\left(\sqrt{-x}\right)}{\sqrt{-x}} & x<0\\[2ex] 1 & x=0\\[2ex] \dfrac{\tan^{-1}\left(\sqrt{x}\right)}{\sqrt{x}} & x>0\end{cases} \tag{4}$$

(Battin refers to the form where $x > 0$ several times in his derivations, but a direct implementation of such for all x obviously leads to incorrect results.) Curiously, one can now readily identify the underlying function in power series form:

$$\begin{aligned} ATANR(x) &= 1 - \operatorname{sgn}(x)\frac{|x|}{3} + \operatorname{sgn}^2(x)\frac{|x|^2}{5} - \operatorname{sgn}^3(x)\frac{|x|^3}{7} + \cdots \\ &= 1 + \sum_{n=1}^{\infty}(-1)^n \operatorname{sgn}^n(x)\frac{|x|^n}{2n+1} \end{aligned} \tag{5}$$

where sgn(x) represents the classical signum function returning the sign of x as (-1,0+1) for x < 0, x = 0 and x > 0, respectively, and $|x|$ represents the absolute value of x.

With the ATANR(x) function defined, one can now use it to evaluate Battin's series, equation (1), by noting that

$$ATANR(x) = \cfrac{1}{1 + \cfrac{x}{3 + \cfrac{4x}{\xi(x)}}} \tag{6}$$

Rearranging and solving for $\xi(x)$, we obtain

$$\xi(x) = \frac{4x\left(1 - ATANR(x)\right)}{(3+x)ATANR(x) - 3} \tag{7}$$

One can clearly see that this representation has a discontinuity at x = 0, but fortunately it is removable. From equation (1), and when partitioned in the same manner as ATANR(x), one obtains

$$\xi(x)=\begin{cases}\dfrac{4x\left(1-ATANR(x)\right)}{(3+x)ATANR(x)-3} & when\, x\neq 0\\ 5 & when\, x=0\end{cases} \tag{8}$$

II.Discontinuity Rectification

The computational accuracy for both the $ATANR(x)$ and $\xi(x)$ functions, equations (4) and (8), respectively, is well within expected square root of machine precision over the vast majority of the domain $x\in(-1,\infty)$. But, equations (4) and (8) both present potential computational difficulties around their respective point discontinuities. In particular, the computation of $\xi(x)$ as $x\to 0^-$ leads to rather gross error as illustrated in figure 1.

Fortunately, thanks to Battin's continued fractions in the form of equations (2) and (1), these difficulties can be readily overcome. One can expand the partition's width near the local discontinuity and evaluate the appropriate function using the first two partial convergents of the true representation to "approximate" their behavior. --In this manner, "approximate" is really an extremely conservative overstatement over the domain width considered about the discontinuity; convergence of the series (1) and (2) is so rapid that the resultant bilinear rational expression from the first two convergents are sufficient.

Therefore, the partitioned representations (4) and (8) can be modified so that observed computational error near the vicinity of the square root of machine tolerance caries across their representative discontinuities:

$$ATANR(x)=\begin{cases}\dfrac{\tanh^{-1}\left(\sqrt{-x}\right)}{\sqrt{-x}} & x<-1.0E-4\\[2ex] \dfrac{15+4x}{15+9x} & |x|<1.0E-4\\[2ex] \dfrac{\tan^{-1}\left(\sqrt{x}\right)}{\sqrt{x}} & x>1.0E-4\end{cases} \tag{9}$$

$$\xi(x) = \begin{cases} \dfrac{4x\left(1 - ATANR(x)\right)}{(3+x)\,ATANR(x) - 3} & when\,|x| > 1.0E-4 \\ \dfrac{315 + 161x}{63 + 16x} & when\,|x| \le 1.0E-4 \end{cases} \tag{10}$$

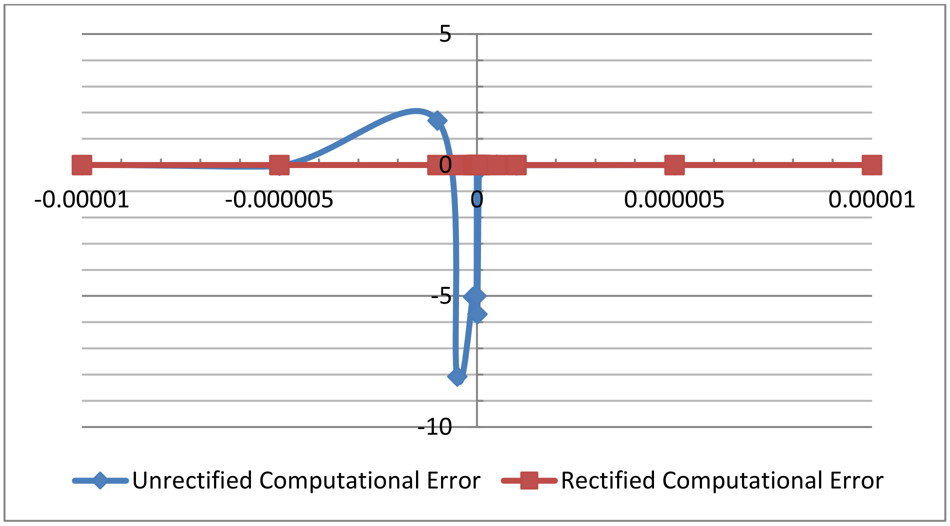

Figure 1 Xi(x) Error Near the Removable Discontinuity.

(Both the unmodified function of equation (8) and the rectified function of equation (10) are shown.)

With these functions now defined, the following Lambert-Battin algorithm is proposed.

III.The Battin-Lambert Algorithm

Given:

- The central body's gravitational constant, μ ,
- The initial position vector, $\vec{r}_0$,
- The final position vector, $\vec{r}_f$,
- A transfer time, Δt , and
- An absolute tolerance on the acceptable root of x, ε .

Determine:

- The corresponding initial and final impulsive delta-v's, $\Delta\vec{v}_0$ and $\Delta\vec{v}_f$, so that a position transfer about the central body can occur with a duration Δt .

Step 1: Determine forms of the transfer angle, $\delta\nu$

$$\begin{aligned}
\cos\delta\nu &= \hat{r}_0 \cdot \hat{r} \\
\sin\delta\nu &= -\sqrt{1-\cos^2\delta\nu} \\
\tfrac{1}{2}\delta\nu &= \tan^{-1}\left(\frac{\sin\delta\nu}{1+\cos\delta\nu}\right) = \tan^{-1}\left(\frac{1-\cos\delta\nu}{\sin\delta\nu}\right) \\
\cos^2\left(\tfrac{1}{4}\delta\nu\right) &= 1/2\left(1+\cos\tfrac{1}{2}\delta\nu\right) \\
\sin^2\left(\tfrac{1}{4}\delta\nu\right) &= 1/2\left(1-\cos\tfrac{1}{2}\delta\nu\right)
\end{aligned} \tag{11}$$

- The hat operation, $\hat{\cdot}$, makes a unit vector in the direction of the vector $\vec{\cdot}$, and
- The total transfer angle $\delta\nu = \left\|2\cdot\tfrac{1}{2}\delta\nu\right\|_{(0,2\pi]}$, which is properly phased so that

Note also: $\delta\nu = \text{atan2}\left(\sin\delta\nu, \cos\delta\nu\right)$ if atan2(y,x) is available

Step 2: Compute auxiliary quantities,

$$
\begin{aligned}
c &= \sqrt{r_0^2 + r^2 - 2r_0 r \cos\delta\nu} \\
s &= 1/2\left(r_0 + r + c\right) \\
R &= r/r_0\,,\ \rho = R - 1 \\
\tan^2 2\omega &= \frac{\rho^2/4}{\sqrt{R} + R\left(2 + \sqrt{R}\right)} \\
r_{0p} &= \sqrt{r_0 r}\left(\cos^2 \tfrac{1}{4}\delta\nu + \tan^2 2\omega\right)
\end{aligned}
\tag{12}
$$

$$
\begin{aligned}
\ell &= \begin{cases} \dfrac{\sin^2 \frac{1}{4}\delta\nu + \tan^2 2\omega}{\sin^2 \frac{1}{4}\delta\nu + \tan^2 2\omega + \cos\frac{1}{2}\delta\nu} & if\ 0 < \delta\nu \le \pi \\ \dfrac{\cos^2 \frac{1}{4}\delta\nu + \tan^2 2\omega - \cos\frac{1}{2}\delta\nu}{\cos^2 \frac{1}{4}\delta\nu + \tan^2 2\omega} & if\ \pi < \delta\nu \le 2\pi \end{cases} \\
m &= \frac{\mu\left(\Delta t\right)^2}{8 r_{0p}^3}
\end{aligned}
\tag{13}
$$

where $r_0 = \left\|\vec{r}_0\right\| = \sqrt{\vec{r}_0 \cdot \vec{r}_0}$ is the norm of the initial position vector, and similarly for the final position vector norm, $r = \left\|\vec{r}\right\| = \sqrt{\vec{r} \cdot \vec{r}}$.

Step 3: Determine *x* and *y*

- Initialize $x_0 = \begin{cases} \ell,\ if\ 0 \le e_0 < 1 \\ 0,\ if\ e_0 \ge 1\ \left(or\ if\ e_0\ unknown\right) \end{cases}$
- Compute new x_{i+1} from x_i for $i = 0,1,2,\cdots$ until $\left|x_{i+1} - x_i\right| \le \varepsilon$:

$$\xi(x)=\begin{cases}\dfrac{4x\left(1-ATANR(x)\right)}{(3+x)\,ATANR(x)-3} & when\,|x|>1.0E-4\\[2ex] \dfrac{315+161x}{63+16x} & when\,|x|\le 1.0E-4\end{cases}$$

where

$$ATANR(x)=\begin{cases}\dfrac{\tanh^{-1}\left(\sqrt{-x}\right)}{\sqrt{-x}} & x<-1.0E-4\\[2ex] \dfrac{15+4x}{15+9x} & |x|<1.0E-4\\[2ex] \dfrac{\tan^{-1}\left(\sqrt{x}\right)}{\sqrt{x}} & x>1.0E-4\end{cases} \qquad (14)$$

- Acquire largest (real) root of the cubic $y^3-(h_1+1)y^2-h_2=0$ where

$$h_1=\frac{(\ell+x)^2(1+3x+\xi)}{(1+2x+\ell)\left(4x+\xi(3+x)\right)} \quad h_2=\frac{m(x-\ell+\xi)}{(1+2x+\ell)\left(4x+\xi(3+x)\right)} \qquad (15)$$

$$x_{i+1}=\sqrt{\tfrac{1}{4}(1-\ell)^2+m/y^2}-\tfrac{1}{2}(1+\ell)$$

Step 4: Determine f, g and

- Determine $a=\dfrac{\mu(\Delta t)^2}{16r_{0p}^2xy^2}$
- When $a>0$:[3]

[3] Alternately and perhaps more clearly, $\alpha=2\sin^{-1}\left(\sqrt{s/2a}\right)$ and if $\Delta t>t_{\min}$ then $\alpha=2\pi-\alpha$

$$
\begin{aligned}
\beta &= \pm 2\sin^{-1}\left(\sqrt{(s-c)/2a}\right) \text{ (+ when } \delta\nu \le \pi\text{, or - if } \delta\nu > \pi\text{)} \\
t_{\min} &= \sqrt{a^3/8\mu}\left(\pi - \beta + \sin\beta\right) \\
\alpha &= \pi \pm 2\cos^{-1}\left(\sqrt{s/2a}\right) \text{ (+ when } \Delta t > t_{\min}\text{, or - if } \Delta t \le t_{\min}\text{)} \\
\Delta E &= \alpha - \beta \\
f &= 1 - a/r_0\left(1 - \cos\Delta E\right) \\
g &= \Delta t - \sqrt{a^3/\mu}\left(\Delta E - \sin\Delta E\right) \\
\dot{g} &= 1 - a/r\left(1 - \cos\Delta E\right)
\end{aligned} \tag{16}
$$

- When $a < 0$:

$$
\begin{aligned}
\alpha &= 2\sinh^{-1}\left(\sqrt{s/-2a}\right) \\
\beta &= 2\sinh^{-1}\left(\sqrt{(s-c)/-2a}\right) \\
\Delta H &= \alpha - \beta \\
f &= 1 - a/r_0\left(1 - \cosh\Delta H\right) \\
g &= \Delta t - \sqrt{|a|^3/\mu}\left(\sinh\left(\Delta H\right) - \Delta H\right) \\
\dot{g} &= 1 - a/r\left(1 - \cosh\Delta H\right)
\end{aligned} \tag{17}
$$

Step 5: Compute the required initial and final velocities

$$
\begin{aligned}
\vec{v}_0^{\,+} &= \left(\vec{r} - f\cdot\vec{r}_0\right)/g \\
\vec{v}_f^{\,-} &= \left(\dot{g}\cdot\vec{r} - \vec{r}_0\right)/g
\end{aligned} \tag{18}
$$

IV.Examples

For comparison, both versions of the Battin-Lambert algorithm are implemented as MATLAB scripts, one evaluating the continued fraction using Lentz's Method (Section 5.2 of Numerical Recipes in C: The Art of Scientific Computing) and the other leveraging intrinsic functions to evaluate ATANR(x). The scripts also compute the number of iterations needed for convergence and execution time using MATLAB's "tic toc" functions.

The first comparison made, using Example 7-5 of "Fundamentals of Astrodynamics and Applications" by Vallado, is for an elliptical transfer. Table 1 (below) shows the results from implementing each algorithm are identical, except for the execution time. The ATANR(x) algorithm is slightly faster by 0.22 of a millisecond. This is attributed to the fact that the continued fraction (inner loop) needs to be evaluated until convergence. In addition, the outer loop ($\left|x_{i+1} - x_i\right| \le 10^{-10}$), required one less iteration.

Table 1 Comparison of Lentz's Method versus ATANR(x) for an Elliptic Transfer

Ellipse	Continued Fractions	ATANR(x)
Initial Velocity (km/s)	{ 2.0586, 2.9158, 0}	{ 2.0589, 2.9160, 0}
Final Velocity (km/s)	{-3.4512, 0.9104, 0}	{-3.4516, 0.9103, 0}
Continued Fractions Iterations	7	N/A
Lambert Iterations	5	4
Computation Time (ms)	0.76	0.54

The comparison of a hyperbolic transfer is made from Example 7-12 of "An Introduction to the Mathematics and Methods of Astrodynamics" by Battin. Table 2 (below) shows the results from implementing each algorithm are, again, identical. The ATANR(x) algorithm is slightly faster by 0.20 of a millisecond. Once more, the outer loop ($\left|x_{i+1} - x_i\right| \le \varepsilon$) required a lesser number of iterations.

Table 2 Comparison of Lentz's Method versus ATANR(x) for a Hyperbolic Transfer

Hyperbola	Continued Fractions	ATANR(x)
Initial Velocity (km/s)	{ -44.1035, 14.3097, 7.2831}	{ -44.1035, 14.3097, 7.2831}
Final Velocity (km/s)	{-45.0875, 8.9540, 6.7380}	{-45.0875, 8.9540, 6.7380}
Continued Fractions Iterations	7	N/A
Lambert Iterations	4	3
Computation Time (ms)	0.62	0.42

V.Conclusions

Notionally, one can see that the generated results from these examples are identical, albeit a slight performance improvement is also realized with the ATANR(x) method. However, to emphasize, the real advantage of implementing ATANR(x) is for simpler, more reliable code implementation, i.e. eliminating the need for loops and convergence tolerance testing.

The four MATLAB scripts (continued fractions versus ATANR(x) for elliptic and hyperbolic transfers) are readily available by contacting the authors at aerospace@cfl.rr.com.

References

Battin, R., "An Introduction to the Mathematics and Methods of Astrodynamics," 2nd printing, AIAA Education, 1987.

Vallado, D., "Fundamentals of Astrodynamics and Applications," 2nd ed., 2nd printing, Space Technology Library, 2004.

W.H. Press, et al., "Numerical Recipes in C: The Art of Scientific Computing," 2nd ed., 5.2 Manipulating Continued Fractions.

Dedication

This Engineering Note is dedicated to the memory of our late friend and colleague, Warren V. Garrison.