High- and Low-thrust Lambert Targeters

High-thrust (classic) Lambert Targeter

[Lancaster and Blanchard(1969)] formulated a unified form of Lambert's theorem already in 1969. In their paper they also made a few suggestions about how to make an algorithm with their findings, but their suggested method still required much user-involvement.
Two decades later, [Gooding(1990)] published a complete procedure implementing their method. His procedure enabled fully automated solutions without any need for manual tweaking, which makes it stand out from for example, the Universal Variables method [Vallado(2001)]. The procedure only requires that a root be found of an analytic function, which makes it also considerably more efficient than for example, Battin's method [Battin(1999)].
Gooding published this complete procedure as FORTRAN-77 code, which was (and still is) widely used in the field of orbit design and optimization. `lamberthigh.m`, one of the two files in this toolbox, is a complete and user-friendly Matlab implementation of his procedure.

`lamberthigh.m`
Usage:

```
[V1, V2] = lamberthigh(r1, r2, tf)
[V1, V2] = lamberthigh(r1, r2, tf, m)
[V1, V2] = lamberthigh(r1, r2, tf, m, mu)
[V1, V2] = lamberthigh(r1, r2, tf, m, mu, way)
```

Dependencies:
`halley.m`

`[V1, V2] = lamberthigh(r1, r2, tf)` finds the conic section that goes through the departure position `r1` = $[x_1, \; y_1, \; z_1]$ and the target position `r2` = $[x_2, \; y_2, \; z_2]$, taking a time `tf` to do so. The results are the two so-called terminal velocity vectors `V1` and `V2`, which are the velocities in this conic section at the departure and target points, respectively.
By default, the departure- and target vectors are expected to be in [km], and the transfer time in [s]. The terminal velocities are also in [km/s].
For example,

```
[V1, V2] = lamberthigh([1, 0, 0]*150e6, [0, 1, 0]*150e6, 365*86400/4)
```

results in

```
V1(:,:,1) =
```

```
  -0.134681252611695   29.812157559007780                        0

V2(:,:,1) =

 -29.812157559007780    0.134681252611695                        0
```

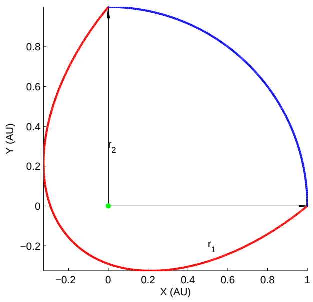
which is (approximately) the Earth's orbital velocity at the two indicated points, in [km/s].

`lamberthigh(r1, r2, tf, m)` will solve the multi-revolution problem. In other words, a total of m complete revolutions are completed before arrival at `r2`. In this case, no solution might exist. In that case, the result will be `NaN`.

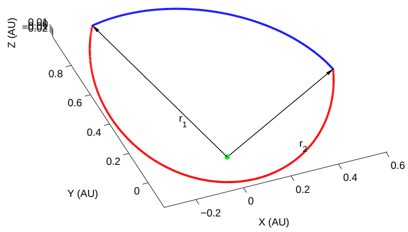
`lamberthigh(r1, r2, tf, m, mu)` will take the value of `mu` as the standard gravitational parameter of the central body. If empty or omitted, the default value is 132712439940, which is the standard gravitational parameter of the Sun, in [km^{3}s-2]. This allows the user to change the central body to any other desired body.
This parameter also allows the user to change the default system of units. For example, if a value of `mu` is used in units of [AU^{3}s-2], also the position vectors `r1` and `r2` will be expected in [AU]. Note that in this case, also the resulting velocities will be in [AU/s].
`lamberthigh(r1, r2, tf, m, mu, way)` allows the user to change the 'way' of the solution. If omitted or empty, both the short-way (`V1(:,:,1), V2(:,:,1)`) and long-way (`V1(:,:,2), V2(:,:,2)`) solutions to the problem are output. This string argument may be equal to `'short'`, in which case only the short-way solution will be returned, or `'long'`, in which case the long-way solution will result. The default is `'both'`, e.g., both solutions.

As a validation (and demonstration) of the proper functioning of this function, four different test cases have been tried, each showing a different feature of this function. The first case is the Earth-Earth example given previously. The second example is a transfer from the Earth to Venus, with Earth-departure on Jan-01-2010 and arrival at Venus on Mar-17-2010, showing the algorithm is fully 3-D compatible. The third example is a direct transfer from Earth to Jupiter, The departure and arrival dates (Jan-01-2015 and Jan-10-2016, respectively) have been selected to be quite unfavorable for such a transfer, to show that hyperbolic trajectories are also not a problem. The last example is a transfer from Earth to Mars, with one full revolution between departure (Mar-11-2005) and arrival (Aug-22-2009). This shows that also such problems are tackled automatically and without any problems.

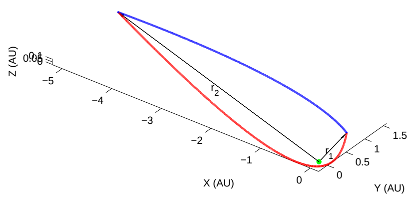
The results for these four cases are shown in Figure 1. Note that both the long-and short ways are plotted for each test case.



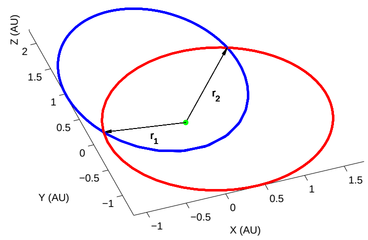
(a) $r_1 = [1.00,\ 0.00,\ 0.00]$ AU
$r_2 = [0.00,\ 1.00,\ 0.00]$ AU
$t_f = \frac{1}{4}$ Year
$m = 0.$



(b) $r_1 = [-0.26,\ 1.45,\ -0.00] \cdot 10^8$km
$r_2 = [9.06,\ 5.92,\ -0.44] \cdot 10^7$km
$t_f = 75$ Days.
$m = 0$



(c) $r_1 = [-0.26,\ 1.45,\ -0.00] \cdot 10^8$km
$r_2 = [-7.77,\ 2.29,\ 0.16] \cdot 10^8$km
$t_f = 374$ Days.
$m = 0$



(d) $r_1 = [-1.47,\ 0.25,\ -0.00] \cdot 10^8$km
$r_2 = [1.40,\ 1.70,\ 0.00] \cdot 10^8$km
$t_f = 1625$ Days.
$m = 1$

Figure 1: Various test cases for the high-thrust heliocentric Lambert-targeter. The blue line shows the short-way solution, and the red line depicts the long-way solution.

Low-thrust (ExpoSin) Lambert Targeter

A.E. Petropoulos explored the possibility of using analytic representations of spirals in the analysis of low-thrust trajectories for his Ph.D. research. His findings, published in this dissertation which he completed in 2001 (see [Petropoulos and Longuski(2004)], although this is a publication based on it), show that a spiral trajectory described by an exponential sinusoid of the form

$$r = k_0 e^{k_1 \sin(k_2\theta+\phi)}$$

is very well suited for this purpose. The equations associated with such trajectories can only be kept in closed-form when tangential thrust is assumed, which makes exponential sinusoids suited for first-order analysis only. In 2006, Dario

Izzo of the ESA Advanced Concepts Team derived the equations for the associated Lambert's problem for exponential sinusoids [Izzo(2006)]. This procedure has been implemented in the second part of this small toolbox, `lambertlow.m`.
`lambertlow.m`
Usage:

```
[V1, V2, exposin] = lambertlow(r1, r2, tf)
[V1, V2, exposin] = lambertlow(r1, r2, tf, k2)
[V1, V2, exposin] = lambertlow(r1, r2, tf, k2, N)
[V1, V2, exposin] = lambertlow(r1, r2, tf, k2, N, mu)
[V1, V2, exposin] = lambertlow(r1, r2, tf, k2, N, mu, way)
```

Dependencies:
`newtonraphson.m`
`regulafalsi.m`

Note that the existence conditions for solutions are a lot stricter for an exponential sinusoid, so that in many cases no solution exists. In those cases, the result will be `NaN`.
`[V1, V2, exposin] = lambertlow(r1, r2, tf)` finds the exponential sinusoid that goes through the departure position `r1 = [`x_1`, `y_1`, `z_1`]` and the target position `r2 = [`x_2`, `y_2`, `z_2`]`, taking a time `tf` to do so. The results are the two so-called terminal velocity vectors `V1` and `V2`, which are the velocities at the departure and target point, respectively.
The `exposin` is the collection of parameters for the exponential sinusoid, in the format `exposin = [k0, k1, k2, phi, tf, N, dth, gamma1, gammam, gammaM]`. The parameters `tf, N, dth, gamma1, gammam,` and `gammaM` (see [Izzo(2006)] for a definition) are appended for completeness or further analysis. The only parameters actually required for the exponential sinusoid are `k0, k1, k2`, and `phi`.

`lamberthigh(r1, r2, tf, k2)` will perform the procedure using a value of `k2` (the only independent variable–the winding parameter). By default, this value is taken equal to $\frac{1}{12}$.

`lamberthigh(r1, r2, tf, k2, N)` will solve the multi-revolution problem. In other words, a total of N complete spiral revolutions are completed before arrival at `r2`.

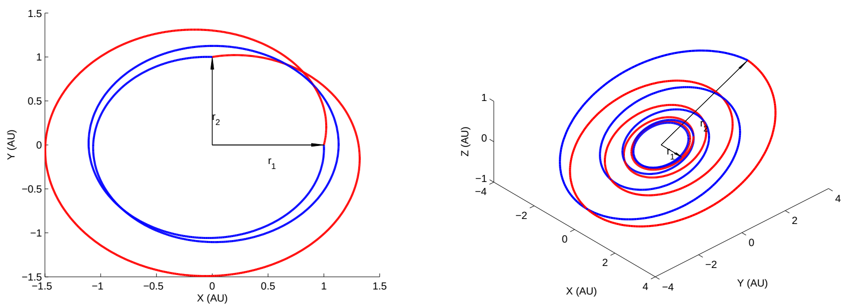
`lambertlow(r1, r2, tf, k2, N, mu)` will take the value of `mu` as the standard gravitational parameter of the central body. If empty or omitted, the default value is 132712439940, which is the standard gravitational parameter of the Sun, in [km^{3}s-2]. This allows the user to change the central body to any other desired body.

This parameter also allows the user to change the default system of units. For example, if a value of mu is used in units of [AU^{3}s−2], also the position vectors r1 and r2 will be expected in [AU]. Note that in this case, also the resulting velocities will be in [AU/s].

`lambertlow(r1, r2, tf, k2, N, mu, way)` allows the user to change the 'way' of the solution. If omitted or empty, both the short-way (`V1(:,:,1), V2(:,:,1), exposin(:,:,1)`) and long-way (`V1(:,:,2), V2(:,:,2), exposin(:,:,2)`) solutions to the problem are output. This string argument may be equal to `'short'`, in which case only the short-way solution will be returned, or `'long'`, in which case the long-way solution will result. The default is `'both'`, e.g., both solutions.

As a validation (and demonstration) of the proper functioning of this function, two different test cases have been tried, showing a few different features of this function. The first case is again the simple Earth-Earth transfer, taking 2 years and 1 complete revolution. The second example is a random outbound trajectory from somewhere around the Earth to an out-of-plane target at about 5 AU, with 4 revolutions revolution and a transfer time of 12 years.

The results for these two cases are shown in Figure 2. Note that both the long-and short ways are plotted for the these test cases.



(a) $r_1 = [1.00,\ 0.00,\ 0.00]$ AU
$r_2 = [0.00,\ 1.00,\ 0.00]$ AU
$t_f = 2$ Years
$N = 1$.

(b) $r_1 = [1.00,\ 0.00,\ 0.00]$ AU
$r_2 = [0.00,\ 4.00,\ 1.00]$ AU
$t_f = 12$ Years
$N = 4$.

Figure 2: Two test-cases for the low-thrust Lambert targeter.

References

[Battin(1999)] R.H. Battin. *An Introduction to the Mathematics and Methods of Astrodynamics, Revised Edition.* American Institute of Aeronautics and Astronautics, Virginia, 1999.

[Gooding(1990)] R.H. Gooding. A procedure for the solution of lambert's orbital boundary-value problem. *Celestial Mechanics and Dynamical Astronomy*, 48:145–165, 1990.

[Izzo(2006)] D. Izzo. Lambert's problem for exponential sinusoids. *Journal of Guidance Control and Dynamics*, 29:1242–1245, September 2006.

[Lancaster and Blanchard(1969)] E.R. Lancaster and R.C. Blanchard. A unified form of lambert's theorem. *NASA technical note TN D-5368*, 1969.

[Petropoulos and Longuski(2004)] A.E. Petropoulos and J.M. Longuski. Shape-based algorithm for automated design of low-thrust, gravity-assist trajectories. *Journal of Spacecraft and Rockets*, 41:787–796, 2004.

[Vallado(2001)] D.A Vallado. *Fundamentals of Astrodynamics and Applications*. Microcosm Press and Kluwer Academic Publishers, second edition, 2001. with contributions by W.D. McCain.