

ENCKE-BETA PREDICTOR FOR ORION BURN TARGETING AND GUIDANCE

Shane Robinson,^{*} Sara Scarritt,^{*} and John L. Goodman[†]

The state vector prediction algorithm selected for Orion on-board targeting and guidance is known as the Encke-Beta method. Encke-Beta uses a universal anomaly (beta) as the independent variable, valid for circular, elliptical, parabolic, and hyperbolic orbits. The variable, related to the change in eccentric anomaly, results in integration steps that cover smaller arcs of the trajectory at or near perigee, when velocity is higher. Some burns in the EM-1 and EM-2 mission plans are much longer than burns executed with the Apollo and Space Shuttle vehicles. Burn length, as well as hyperbolic trajectories, has driven the use of the Encke-Beta numerical predictor by the predictor/corrector guidance algorithm in place of legacy analytic thrust and gravity integrals.

INTRODUCTION

On-board software tasks for on-orbit burn targeting, guidance, and navigation require the use of trajectory integrators to predict state vectors forward or backward in time. Burn targeting methods and active guidance routines make extensive use of trajectory prediction algorithms. These predictions may be over time intervals ranging from tens of seconds to hours or even days. Such predictions require higher fidelity numerical integration schemes and higher order acceleration models than very short time interval propagations cyclically executed at high rates, such as those found in the propagation phase of an on-board Kalman filter. A high fidelity state vector predictor may also be used to predict a state vector uplinked by Mission Control forward or backward in time to the current on-board time, so that the high rate short time interval propagator can be reinitialized.

In 1852 Johann Franz Encke (1791-1865) published a new method for integrating a trajectory in cases where the disturbing acceleration (higher order gravitational acceleration from the primary body, and gravitational accelerations from secondary bodies) was small compared to the central body point mass (conic) gravitational acceleration. [1, 2, 3] This method was particularly useful in the days before digital computers were available for trajectory computations.[‡] For example, in his original 1852 paper, Encke claims to have only required 11 hours (including distractions and breaks) to calculate 13 data points of Vesta's perturbed orbit - a task that had previously been impossible in so short a time frame. In Encke's original formulation, Encke-Time, the differential equations are integrated with respect to time. During the ensuing 160 years Encke-Time has been and continues to be used for a wide array of applications requiring the rapid integration of perturbed trajectories. Encke's methods are particularly popular in the crewed spacecraft community. Another version of

^{*} Aerospace Engineer, Aeroscience and Flight Mechanics Division, NASA Johnson Space Center, 2101 NASA Parkway, Houston TX 77058.

[†] Specialist Engineer, Odyssey Space Research, LLC., 1120 NASA Parkway, Suite 505 Houston, TX 77058.

[‡] Encke's method was developed at least 56 years before Cowell published his method in 1908. [4]

Encke that used a universal variable called beta as the independent variable, Encke-Beta, was in use in the 1960s. [5] The origins of Encke-Beta are not as well documented as Encke-Time. Battin's 1964 book briefly mentions the use of a universal variable in place of time in a problem for the reader (page 216, problem 6.2). [6] Robert Baker briefly discusses Encke-Beta on pages 241 and 242 of his 1967 book. [7] Encke-Beta was used at Mission Control in Houston in support of Apollo, Skylab, Apollo-Soyuz, and Space Shuttle missions. The Encke-Beta method will also be used by the Orion flight computer during the on-orbit phase of flight. An overview of trajectory predictors used by previous NASA human flight vehicles is given in the Appendix A.

The paper starts with a discussion of how Encke's method will be used by Orion on-board burn targeting and guidance. The theory of the Encke-Time method is then presented. This is followed by a section on regularization via the Sundman transformation. This transformation is a key to understanding the following sections on the Encke-Beta method and the solution of Kepler's equation. Step size selection and stopping criteria for Encke-Beta are discussed. A comparison of Encke-Time and Encke-Beta predictions is provided to assess the accuracies of the two methods. There are three appendices. Appendix A covers the use of trajectory prediction methods in the NASA human flight program. Appendix B covers analytic computation of the Stumpff functions used in solving Kepler's equation. Analytic expression are also included continued fraction expansion used to rapidly compute the Stumpff functions. Appendix C details the evaluation of continued fractions used to compute the Stumpff functions.

USE OF ENCKE'S METHOD BY ORION

The on-board Two Level Targeter (TLT) must perform numerous state vector predictions while targeting burns. [8, 9] These predictions may be performed in Earth or lunar orbit, or in cislunar space where the primary body is either the Earth, Moon, or Sun. Secondary bodies must also be taken into account when performing cislunar trajectory prediction.

The predictor/corrector architecture of Orion Orbit Guidance (OrbGuid) uses burn targets computed by either the TLT or Mission Control at the NASA/Johnson Space Center. [8, 10] In either case, OrbGuid computes the current burn solution using Encke-Beta to determine if the trajectory constraints are met. Any misses that are outside of tolerances are used to adjust the steering parameters. On previous vehicles (Saturn boosters, Apollo Lunar Module, the Space Shuttle) these predictions were performed with analytic thrust and gravity integrals. Avoiding costly numerical integration continues to be important on low capacity flight computers that had to complete guidance processing in a given amount of time. However, this constraint is relaxed on the Orion vehicle as compared to the Saturn and Space Shuttle. Gravity integrals used by Space Shuttle PEG (OrbGuid's predecessor) for orbit insertion and deorbit guidance are incompatible with the hyperbolic trajectories of some proposed exploration missions. [11] The more advanced flight computers of the Orion spacecraft enable numerical integration using Encke-Beta to be performed during a guidance cycle. The advantage of Encke-Beta over some other integration methods, such as variation of parameters, is that it uses a lower number of integration variables. [12, 13]

An additional advantage of the Encke-Beta predictor is that it combines the speed and accuracy benefits of a variable-step propagator with the rigidity of a fixed-step propagator. A fixed Beta step size results in shorter time steps near periape, where the vehicle motion is changing quickly, and longer time steps farther away where the dynamics are less sensitive. This combination is particularly well-suited to the disparate needs of the Orion targeting and guidance algorithms. The TLT requires several iterations of long propagation arcs over fixed time intervals, and the speed

gained through the variable time steps ensures that these computations are completed within the time allotted for targeting. OrbGuid, on the other hand, must complete its calculations within a fixed number of steps in order to operate in its assigned 1 Hz cycle. Encke-Beta satisfies both of these performance requirements, and in trade studies also demonstrated generally better accuracy with fewer integration steps and shorter computation time than other predictor options.

THE ENCKE-TIME METHOD

Encke's method using time as the independent variable is best introduced by contrasting it with Cowell's method. [12, 14, 15] The Cowell technique involves direct integration of the equations of motion in rectangular coordinates. The acceleration includes central force gravity (conic) as well as any perturbative accelerations, such as higher order gravitational harmonics, atmospheric drag, gravitational acceleration from other celestial bodies (Moon, Sun, planets, etc.), solar radiation pressure, etc. Time is the independent variable of integration. Cowell's method is efficient if the order of the perturbation accelerations is approximately equal to or larger than the central force gravity acceleration. [3] The differential equation integrated by Cowell's method is

$$\ddot{\vec{r}} = -\frac{\mu}{r^3}\vec{r} + \vec{a}_p \quad (1)$$

As is customary, here a dot (·) above a parameter represents differentiation with respect to time. Two dots indicate second derivatives. When a_p is zero this becomes the differential equation for a pure conic trajectory

$$\ddot{\vec{r}}_c = -\frac{\mu}{r_c^3}\vec{r}_c \quad (2)$$

Encke's method integrates the perturbation accelerations (any accelerations that are not due to the primary body represented as a point mass, also called central force gravity or conic gravity). A conic-relative position and velocity perturbation state vector $(\vec{\delta}, \vec{\nu})$ is maintained by integrating the perturbation accelerations, while a conic state vector (\vec{r}_c, \vec{v}_c) is maintained using Kepler's equation. This conic state vector is sometimes called the osculating state vector or the osculating trajectory. For clarity it will be referred to as the conic state vector or the conic trajectory, since the state vector represents a conic section (ellipse, circle, parabola, or hyperbola). The conic-relative perturbation state vector $(\vec{\delta}, \vec{\nu})$ and the conic state vector (\vec{r}_c, \vec{v}_c) are computed at the same time tag. The complete or total state vector (\vec{r}, \vec{v}) at a given time may be computed by adding the conic and the conic-relative perturbation state vectors

$$\vec{r} = \vec{r}_c + \vec{\delta} \quad (3)$$

Note that these terms are contemporaneous in the sense that all terms occur at the same time.

The differential equation for the conic-relative perturbation state vector $(\vec{\delta}, \vec{\nu})$ integrated during the Encke-Time method is derived as follows, based on the nomenclature in Battin's 1999 book. Begin by differentiating this equation with respect to time to yield

$$\dot{\vec{r}} = \dot{\vec{r}}_c + \dot{\vec{\delta}} \quad \text{or} \quad \vec{v} = \vec{v}_c + \vec{\nu} \quad (4)$$

differentiating a second time gives

$$\ddot{\vec{r}} = \ddot{\vec{r}}_c + \ddot{\vec{\delta}} \quad (5)$$

Now solve for $\ddot{\delta}$ and substitute in equations 1 and 2

$$\ddot{\delta} = \frac{\mu}{r_c^3} \vec{r}_c - \frac{\mu}{r^3} \vec{r} + \vec{a}_p \quad (6)$$

which can be written as

$$\ddot{\delta} = -\frac{\mu}{r_c^3} \left[\vec{\delta} + \left(\frac{r_c^3}{r^3} - 1 \right) \vec{r} \right] + \vec{a}_p \quad (7)$$

This is the differential equation integrated by the Encke-Time method.

Figure 1 shows how vectors \vec{r}_c , \vec{r} , and $\vec{\delta}$ form a vector triangle. For cases where $\delta \ll r_c$ this geometry gives rise to numerical difficulties when calculations are done on a finite word length computer. This is recognized as the interior problem which has a well known alternative formulation avoiding the numerical difficulties. This alternative formulation relies on a convenient parameter representing the disparity in the squares of r and r_c .

$$q = \frac{r_c^2}{r^2} - 1 = \frac{\vec{\delta} \cdot (\vec{\delta} - 2\vec{r})}{\vec{r} \cdot \vec{r}} \quad (8)$$

This small quantity will be zero when $r = r_c$, positive when $r < r_c$, and negative for $r > r_c$. The triangle inequality can then be used to compute bounds on the value of q

$$\frac{r_c^2}{(r_c + \delta)^2} - 1 \leq q \leq \frac{r_c^2}{(r_c - \delta)^2} - 1 \quad (9)$$

This parameter can be used to find a stable solution to the numerically difficult term in equation 7

$$f = \frac{r_c^3}{r^3} - 1 = \frac{\left[(1+q)^{\frac{3}{2}} - 1 \right] \left[(1+q)^{\frac{3}{2}} + 1 \right]}{(1+q)^{\frac{3}{2}} + 1} = q \frac{3 + 3q + q^2}{(1+q)^{\frac{3}{2}} + 1} \quad (10)$$

The Encke-Time differential equation can now be written in it's final form

$$\ddot{\delta} = -\frac{\mu}{r_c^3} \left[\vec{\delta} + f\vec{r} \right] + \vec{a}_p \quad (11)$$

Encke's method is more efficient than Cowell when the perturbation acceleration is much less than the central force gravity acceleration. Encke can provide the same level of accuracy as Cowell with larger integration step sizes. However, Encke is best suited for prediction when the perturbation state vector is small. To maintain the accuracy of the method the conic state vector is periodically reset to the total state vector, and the perturbation state vector is set to zero. This is known as rectification. Rectification is usually triggered when the ratio of the conic-relative perturbed position vector magnitude (δ) and the conic position vector magnitude (r_c) exceed some threshold, such as $\delta_{thresh}/r_c = 0.001$. The ratio of the conic relative perturbed velocity vector magnitude (ν) to the conic velocity vector magnitude (v_c) is sometimes used as well. However, velocity rectification requires a different threshold. The position threshold has some utility as it can be tied directly the parameters used to solve the interior problem. The position threshold can be used in equation 9 to produce approximate bounds on the value of q .

$$\frac{1}{(1 + \delta_{thresh}/r_c)^2} - 1 - \varepsilon \leq q \leq \frac{1}{(1 - \delta_{thresh}/r_c)^2} - 1 + \varepsilon \quad (12)$$

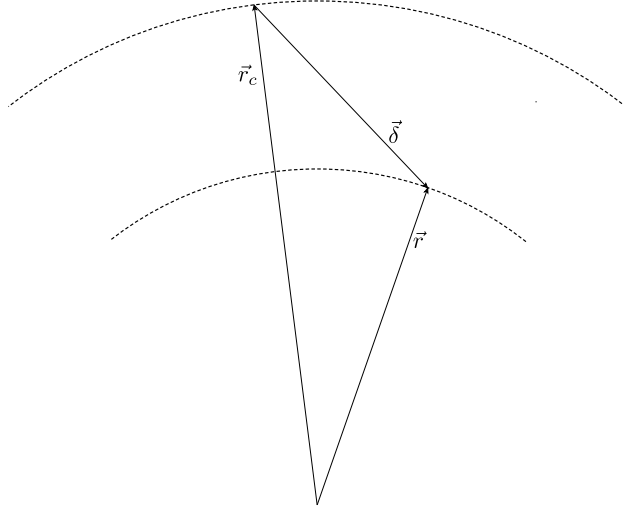


Figure 1. Fundamental geometry for Encke's method.

where ε is a small (relative to q) positive number associated with $\delta > \delta_{thresh}$ just before rectification. Approximate bounds on f can be found by applying the same reasoning to 10. Although not exact, these limits can serve an excellent heuristic on the value of q . For example, given $\delta_{thresh}/r_c = 0.001$ the approximate bounds on q are $-0.001997 - \varepsilon \leq q \leq 0.002003 + \varepsilon$. These bounds on q provide motivation for calculating 10 using a power series. The power series can be more computational efficient for implementations where the evaluation of the square root is costly

$$f = q \frac{3}{2} \left[1 + \frac{\overbrace{(-1)!!}^{\frac{1}{4}}}{2^1 2!} q - \frac{\overbrace{1!!}^{\frac{1}{24}}}{2^2 3!} q^2 + \frac{\overbrace{3!!}^{\frac{1}{64}}}{2^3 4!} q^3 - \frac{\overbrace{5!!}^{\frac{1}{128}}}{2^4 5!} q^4 + \frac{\overbrace{7!!}^{\frac{7}{1536}}}{2^5 6!} q^5 - \dots \right] \quad (13)$$

$$= q \frac{3}{2} \sum_{n=0}^{\infty} (-1)^{n+1} \frac{(2n-3)!!}{2^n (n+1)!} q^n = q \frac{3}{4\sqrt{\pi}} \sum_{n=0}^{\infty} (-1)^{n+1} \frac{\Gamma(n - \frac{1}{2})}{(n+1)!} q^n \quad (14)$$

For the above example ($\delta_{thresh}/r_c = 0.001$) notice that only the terms up to and including $q^{4/128}$ will contribute to accuracy improvement for double precision numbers.

Figure 2 illustrates the Encke-Time process over two rectification cycles. At the beginning, in the far right of the illustration, the total perturbed state vector (\vec{r}, \vec{v}) is used to initialize a conically propagated state vector (\vec{r}_c, \vec{v}_c) (i.e. solving Kepler's equation and computing the Lagrange coefficients). At the same time, the conic-relative perturbation state vector $(\vec{\delta}, \vec{v})$ is integrated from an initial value of zero. Over time the magnitude of the conic-relative perturbation state vector $(\vec{\delta}, \vec{v})$ increases, and eventually the specified threshold is exceeded. A rectification is performed, the new

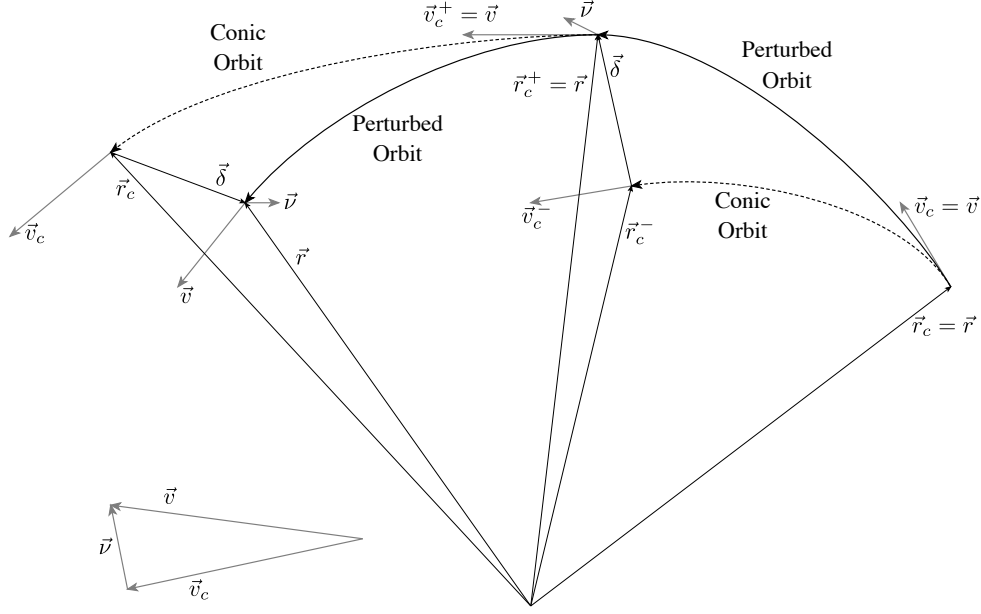


Figure 2. Encke-Time prediction

conic state vector (\vec{r}_c, \vec{v}_c) is set to the sum of the old conic state vector (\vec{r}_c, \vec{v}_c) and the old conic-relative perturbation state vector $(\vec{\delta}, \vec{\nu})$. The conic-relative perturbation state vector $(\vec{\delta}, \vec{\nu})$ is set to zero, and the process repeats.

REGULARIZATION VIA THE SUNDMAN TRANSFORMATION

Before proceeding, it is helpful to briefly review the regularization of Keplerian orbital mechanics via the Sundman transformation. A universal anomaly can be used in place of time as the independent variable. This allows for a single unified version of Kepler's equation for circular, elliptic, parabolic, and hyperbolic orbits. This has the computational benefit of providing a single logical path through the routine. A universal anomaly is a monotonic function of time and is valid for circular, elliptical, parabolic, and hyperbolic orbits. This derivation uses the U_n Stumpff functions presented by Battin on pages 174 to 179 of his 1999 book. [16] Battin uses the universal anomaly χ which is the same as the variable β in this document. Note that primes (') beside a quantity represents differentiation with respect to β in the same way dots represent differentiation with respect to time. The Sundman transformation defines the universal anomaly β with the differential equation

$$\frac{dt}{d\beta} = t' = \frac{r_c}{\sqrt{\mu}} \quad (15)$$

Note that in some of the literature the Sundman transformation is used without the factor $1/\sqrt{\mu}$. [12, 17] The advantage to omitting the factor $1/\sqrt{\mu}$ is that a negative value of μ can be used allowing the same equations to be valid for repulsive forces. Since repulsive forces are not of interest here, the factor $1/\sqrt{\mu}$ is used to cause values of β to be relatable purely in terms of orbital geometry. That is, β is not dependent on μ of the central body. μ of the central body only enters the problem in the mapping from β to time. This desirable property will be shown later.

β is the universal anomaly used in the Orion predictor and is the same anomaly used in the Apollo and Space Shuttle Mission Control software (see the Appendix A). Specifically, β is also the same as Battin's χ (1999 book), Battin's x (1964 book), and Herrick's \hat{X} . [18, 19]

Using β and the U_n Stumpff functions (pages 175 to 179), Battin regularizes the equations for a conic orbit to get the key relations

$$\sigma_c = \sigma_{c_0} U_0(\beta; \alpha_c) + (1 - \alpha_c r_{c_0}) U_1(\beta; \alpha_c) \quad (16)$$

$$r_c = r_{c_0} U_0(\beta; \alpha_c) + \sigma_{c_0} U_1(\beta; \alpha_c) + U_2(\beta; \alpha_c) \quad (17)$$

$$\sqrt{\mu}(t - t_0) = r_{c_0} U_1(\beta; \alpha_c) + \sigma_{c_0} U_2(\beta; \alpha_c) + U_3(\beta; \alpha_c) \quad (18)$$

where

$$\sigma = \frac{\vec{r} \cdot \vec{v}}{\sqrt{\mu}} \quad \text{and} \quad \alpha_c = \frac{1}{a_c} = \frac{2}{r_c} - \frac{v_c^2}{\mu} \quad (19)$$

and the subscript c_0 denotes the conditions from which the conic will be propagated. The quantity σ is defined such that

$$r'_c = \sigma_c \quad (20)$$

The first and second derivative of \vec{r} with respect to β will be required later and are now developed. The first derivative is easily found using the chain rule

$$\vec{r}' = \frac{\vec{v}}{dt} \frac{dt}{d\beta} = \frac{r_c}{\sqrt{\mu}} \vec{v} \quad (21)$$

The second derivative of \vec{r} with respect to β is obtained in a similar fashion through judicious use of the chain rule.

$$\vec{r}'' = \frac{d}{d\beta} \left(\frac{r_c}{\sqrt{\mu}} \vec{v} \right) = \frac{1}{\sqrt{\mu}} \frac{dr_c}{d\beta} \vec{v} + \frac{r_c}{\sqrt{\mu}} \frac{d\vec{v}}{d\beta} = \frac{\sigma_c}{r_c} \vec{r}' + \frac{r_c^2}{\mu} \ddot{\vec{r}} \quad (22)$$

Equation 1 can now be substituted in to give

$$\vec{r}'' = \frac{\sigma_c}{r_c} \vec{r}' + \frac{r_c^2}{\mu} \left(-\frac{\mu}{r^3} \vec{r} + \vec{a}_p \right) = \frac{\sigma_c}{r_c} \vec{r}' - \frac{r_c^2}{r^3} \vec{r} + \frac{r_c^2}{\mu} \vec{a}_p \quad (23)$$

In the case of pure conic motion the equation for the first derivative is simply equation 21 with subscripts c added. The second derivative of \vec{r} with respect to β for conic motion is

$$\vec{r}''_c = \frac{\sigma_c}{r_c} \vec{r}'_c - \frac{1}{r_c} \ddot{\vec{r}}_c \quad (24)$$

Equation 18 is the universal form of Kepler's equation (page 178, equation 4.81). The universal anomaly is equivalent to the following terms for circular, elliptic, parabolic, and hyperbolic orbits respectively

$$\beta = \frac{\Delta M}{\sqrt{\alpha_c}} = \frac{\Delta E}{\sqrt{\alpha_c}} = \sigma - \sigma_{c_0} = \frac{\Delta H}{\sqrt{-\alpha_c}} \quad (25)$$

At this point note that the earlier claim that β would be purely a function of geometry is verified. These relations are only dependent on the semi-major (or semi-transverse) axis with no dependence on μ for the central body. For elliptic and circular orbits the time of one orbital period is

$$P = 2\pi\sqrt{\frac{a^3}{\mu}} = \frac{2\pi}{\sqrt{\mu\alpha_c^3}} \quad (26)$$

From equation 25 it is clear that the universal anomaly associated with one orbital period is

$$\beta_P = \frac{2\pi}{\sqrt{\alpha_c}} \quad (27)$$

Again note that β is not dependent on μ of the central body and is solely a geometric quantity.

THE ENCKE-BETA METHOD

The Encke-Beta method consists of using Encke's approach with the universal anomaly β . As with Encke-Time begin with the relation

$$\vec{r} = \vec{r}_c + \vec{\delta} \quad (28)$$

As with equation 3 these terms are contemporaneous in the sense that all terms occur at the same time. However, the independent variable is now β which is mapped to time through the Sundman transformation applied to the conic trajectory. That is to say, β maps to $t - t_0$ through equation 18.

Proceed by solving for $\vec{\delta}$ and differentiating the result twice with respect to β

$$\vec{\delta}' = \vec{r}' - \vec{r}_c' \quad (29)$$

$$\vec{\delta}'' = \vec{r}'' - \vec{r}_c'' \quad (30)$$

Substituting equations 21, 23, and 24 into these relations gives

$$\vec{\delta}' = \frac{r_c}{\sqrt{\mu}} (\vec{v} - \vec{v}_c) = \frac{r_c}{\sqrt{\mu}} \vec{v} \quad (31)$$

$$\vec{\delta}'' = \frac{\sigma_c}{r_c} \underbrace{(\vec{r}' - \vec{r}_c')}_{\vec{\delta}'} + \frac{r_c^2}{\mu} \underbrace{\left(\frac{\mu}{r_c^3} \vec{r}_c - \frac{\mu}{r_c^3} \vec{r} + \vec{a}_p \right)}_{\ddot{\delta}} \quad (32)$$

Note that the term labeled $\ddot{\delta}$ corresponds to equation 6. For most practical problems the quantity $\frac{r_c}{\sqrt{\mu}} > 1$. It follows directly that \vec{v} and $\vec{\delta}'$ are co-aligned and $\nu < \delta'$. The vectors $\ddot{\delta}$ and $\vec{\delta}''$ generally have distinct directions. Although equation 31 isn't required for integration, it does provide an important mapping from $\vec{\delta}'$ to \vec{v}

$$\vec{v} = \vec{v}_c + \frac{\sqrt{\mu}}{r_c} \vec{\delta}' \quad (33)$$

Following the same procedure outlined in the Encke-Time section leads to the final Encke-Beta differential equation

$$\vec{\delta}'' = \frac{1}{r_c} \left[\sigma_c \vec{\delta}' - \vec{\delta} - f \vec{r} \right] + \frac{r_c^2}{\mu} \vec{a}_p \quad (34)$$

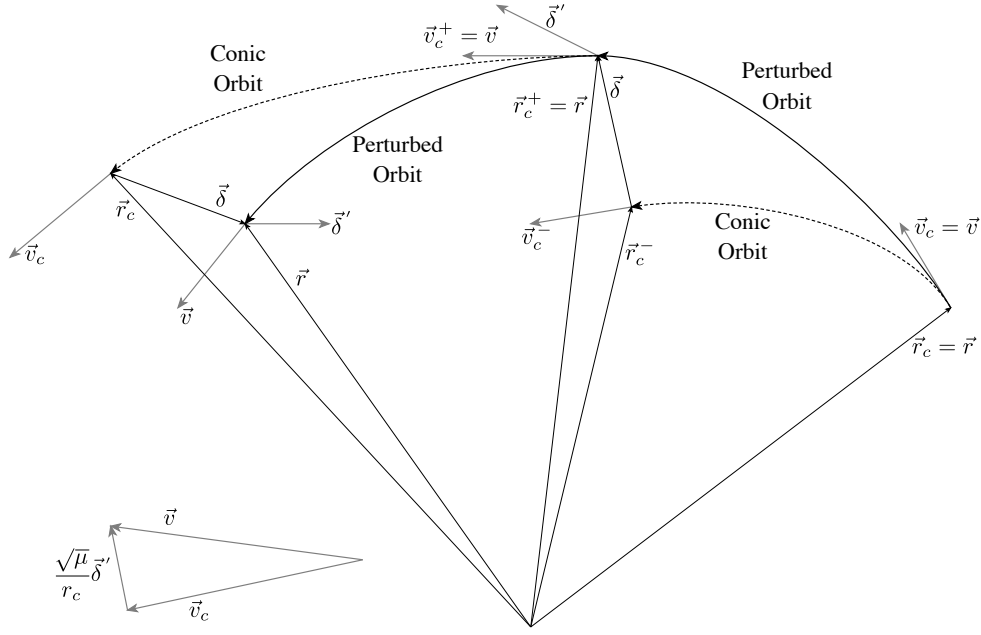


Figure 3. Encke-Beta prediction.

Encke-Beta will use a fourth order Nystöm-Lear (NLZD4/4) integrator that integrates equations of the type $\ddot{z} = f(z, \dot{z}, t)$. [20, 21]

Figure 3 is an illustration of the Encke-Beta process over two rectification cycles. The concepts in the illustration are the same as for Figure 2, except that to compute the total perturbed velocity vector \vec{v} the first derivative of the conic-relative vector $\vec{\delta}'$ is multiplied by a conversion factor before it is added to \vec{v}_c , since it is integrated with respect to the universal variable β .

Figure 4a contains outline of the Encke_Time and Figure 4b contains the Encke-Beta algorithm. The inputs to both algorithms are the initial position and velocity vectors, the time of the initial vector, the time that the state vector is to be predicted to, and the number of integration steps.

Encke_Time computes the Δt for the integration step (2.). The integration loop (step 3) consists of three steps. Rectification is performed if required (3.a.). Then the conic relative perturbed state vector $(\vec{\delta}, \dot{\vec{\delta}})$ is integrated (3.b.). At each the end of each integration step and the conic state vector (\vec{r}_c, \vec{v}_c) is computed (3.b). The total state vector (\vec{r}, \vec{v}) is then computed (3.c.). The process is repeated until all the required number of steps have been integrated.

Before the Encke_Beta integration begins, a Kepler integration with respect to time is performed from the initial time to the final time to estimate the value of β over the prediction interval (2.). This is divided by the number of steps to determine the integration step size, $\Delta\beta$ (3.). Within the integration loop rectification is performed if it is required (4.a.). If rectification was performed another Kepler time prediction to the final time is performed to obtain a more accurate estimate of the β over the remaining prediction interval (4.a.ii). The integration step size is then recomputed (4.a.iii.). While the integration of $(\vec{\delta}', \vec{\delta}'')$ is performed (4.b.) the time at each the end of each integration step the conic state vector (\vec{r}_c, \vec{v}_c) at that time are computed. The total state vector (\vec{r}, \vec{v})

Encke_Time ($\vec{r}_0, \vec{v}_0, t_0, t_f, n_{steps}$)

1. $\vec{r}_{rec} = \vec{r}_0, \vec{v}_{rec} = \vec{v}_0, \vec{\delta} = 0, \vec{v} = 0$.
2. Compute Δt from t_0, t_f , and n_{steps} .
3. Integrate state vector over n_{steps} .
- a. Rectify if needed:
 - i. $\vec{r}_{rec} = \vec{r}_0, \vec{v}_{rec} = \vec{v}_0, \vec{\delta} = 0, \vec{v} = 0$.
 - b. Integrate $\vec{\delta}$ and \vec{v} .
- c. Compute \vec{r} and \vec{v} and $\vec{r}_c, \vec{v}_c, \vec{\delta}$, and \vec{v} computed during integration.*
4. Return \vec{r}, \vec{v} , and t .

* \vec{r}_c and \vec{v}_c are computed during integration when the differential equations are evaluated.

(a) Outline of the Encke_Time Algorithm.

Encke_Beta ($\vec{r}_0, \vec{v}_0, t_0, t_f, n_{steps}$)

1. $\vec{r}_{rec} = \vec{r}_0, \vec{v}_{rec} = \vec{v}_0, \vec{\delta} = 0, \vec{\delta}' = 0$.
2. Compute β_f by performing Kepler time prediction of \vec{r}_0 and \vec{v}_0 from t_0 to t_f .
3. Compute $\Delta\beta$ from β_f and n_{steps} .
4. Integrate state vector over n_{steps} .
- a. Rectify if needed:
 - i. $\vec{r}_{rec} = \vec{r}_0, \vec{v}_{rec} = \vec{v}_0, \vec{\delta} = 0, \vec{v} = 0$.
 - ii. Compute β_f by performing Kepler time prediction of \vec{r}_{rec} and \vec{v}_{rec} from t to t_f .
 - iii. Compute new $\Delta\beta$ from β_f and remaining number of steps.
- b. Integrate $\vec{\delta}'$ and $\vec{\delta}''$.
- c. Compute \vec{r} and \vec{v} using \vec{r}_c, \vec{v}_c , and t with $\vec{\delta}$, and $\vec{\delta}'$ computed during integration.[†]
- d. Compute remaining number of steps.
5. Return \vec{r}, \vec{v} , and t .

[†] \vec{r}_c, \vec{v}_c , and t are computed during integration when the differential equations are evaluated.

(b) Outline of the Encke_Beta Algorithm.

Figure 4. Outlines of the Encke algorithms.

is recomputed (4.c.), along with the remaining number of integration steps (4.d.). Execution of the loop continues until the specified number of steps have been taken.

SOLVING KEPLER'S PROBLEM

Both the Encke-Beta and Encke-Time algorithms require the solution of Kepler's problem with respect to time. Shepperd's algorithm was chosen since it is efficient and can easily handle elliptical, circular, parabolic, and hyperbolic orbits. [17] It can also compute a conic state transition matrix. Battin has a different presentation of the same method on pages 219 and 220 of his 1999 book. [3] Here the key equations from Battin are presented without development. Define the new variable

$$w = \frac{U_1\left(\frac{\beta}{4}; \alpha_c\right)}{U_0\left(\frac{\beta}{4}; \alpha_c\right)} = \frac{U_2\left(\frac{\beta}{2}; \alpha_c\right)}{U_1\left(\frac{\beta}{2}; \alpha_c\right)} = \frac{\frac{1}{4}\beta}{1 - \frac{\alpha_c\left(\frac{1}{4}\beta\right)^2}{3 - \frac{\alpha_c\left(\frac{1}{4}\beta\right)^2}{5 - \frac{\alpha_c\left(\frac{1}{4}\beta\right)^2}{7 - \ddots}}} \quad (35)$$

if $|\beta| < \pm\beta_P/2 = \pi/\sqrt{\alpha_c}$ for closed orbits, observe that $|w| \leq 1/\sqrt{|\alpha_c|}$. By leveraging the properties of the Stumpff functions it can be shown that

$$U_0(\beta; \alpha_c) = 1 - 8z(1 - z) \quad (36)$$

$$U_1(\beta; \alpha_c) = 4(1 - 2z)(1 - z)w \quad (37)$$

$$U_2(\beta; \alpha_c) = 8(1 - z)^2 w^2 \quad (38)$$

$$U_3(\beta; \alpha_c) = \alpha_c \mu W + \frac{1}{3} U_1(\beta; \alpha_c) U_2(\beta; \alpha_c) \quad (39)$$

where

$$z = \frac{\alpha_c w^2}{1 + \alpha_c w^2} \quad (40)$$

For elliptic orbits $z \in [0, 1/2]$ with $z = 0$ corresponding to $t = t_0$ ($\beta = 0$) and $z = 1/2$ corresponding to $t = t_0 + \frac{2\sigma_{c0}}{\alpha_c \sqrt{\mu}} \pm P/2$ ($\beta = \pm\beta_P/2 = \pm\pi/\sqrt{\alpha_c}$). For parabolic orbits $z = 0$, and for hyperbolic orbits $z \in [-\infty, 0]$. The function W is given by

$$W = \frac{512}{15} [(1 - z)w]^5 G\left(5, 0, \frac{5}{2}, z\right) + \Delta W \quad (41)$$

where $G\left(5, 0, \frac{5}{2}, z\right)$ is a Gaussian continued fraction which can be rapidly computed with the continued fraction*

$$G\left(5, 0, \frac{5}{2}, z\right) = \frac{1}{1 - \frac{\frac{10}{7}z}{1 + \frac{\frac{2}{21}z}{1 - \frac{\frac{28}{33}z}{1 + \dots}}}} \quad (42)$$

The quantity ΔW accounts for multiple revolutions about a closed orbit

$$\Delta W = \begin{cases} 0 & \alpha_c \leq 0 \\ n \frac{2\pi}{\sqrt{\alpha_c^5}} & \alpha_c > 0 \end{cases} \quad (43)$$

where

$$n = \left\lfloor \frac{t_0 + \frac{P}{2} - \frac{2\sigma_{c0}}{\alpha_c \sqrt{\mu}}}{P} \right\rfloor \quad (44)$$

Equation 18 can now be used to quickly find $t - t_0$ as a function of w . To solve Keplers equation, the first derivative is required

$$\frac{dt}{dw} = 4r(1 - z) \quad (45)$$

And the Newton-Raphson iterate required to solve Kepler's equation is

$$w^+ = w^- - \frac{t - t_0}{4r(1 - z)} \quad (46)$$

*See Appendix C for a discussion concerning the evaluation of continued fractions.

The Lagrange coefficients are now easily computed

$$\vec{r}_c = \left[1 - \frac{1}{r_0} U_2(\beta; \alpha_c) \right] \vec{r}_0 + \frac{1}{\sqrt{\mu}} [r_{c0} U_1(\beta; \alpha_c) + \sigma_{c0} U_2(\beta; \alpha_c)] \vec{v}_0 \quad (47)$$

$$\vec{v}_c = \left[-\frac{\sqrt{\mu}}{r r_0} U_1(\beta; \alpha_c) \right] \vec{r}_0 + \left[1 - \frac{1}{r} U_2(\beta; \alpha_c) \right] \vec{v}_0 \quad (48)$$

For this work it will be important to rapidly evaluate the first few Stumpff functions as a function of β . This can be done using only a single continued fraction evaluation of equation 35 * to find the term

Now the Stumpff functions required to evaluate equations 47 and 48 are easily computed using equations 36-43

Note that this series can be shown to be convergent for all values of β which the only exception being $\beta = 2\pi \pm n4\pi$ and $\alpha_c > 0$. [3] Nevertheless, any potential difficulty may be avoided by simply performing a modulo operation when $\alpha_c > 0$

$$\beta = \left[\left(\beta - \frac{\pi}{\sqrt{\alpha_c}} \right) \mod \frac{2\pi}{\sqrt{\alpha_c}} \right] - \frac{\pi}{\sqrt{\alpha_c}} \quad (49)$$

For Encke-Beta a version of Kepler's equation relating β to time is required in step 4c of Figure 4b. Evaluation of U_3 and the associated continued fraction can be avoided by using equation 52 to find the time

$$\sqrt{\mu}(t - t_0) = \begin{cases} \frac{1}{\alpha_c} \left(\beta - (1 - z) [8\sigma_{c0}z + 4(1 - \alpha_c r_{c0})(1 - 2z)w] \right) & \alpha_c \neq 0 \\ r_{c0}\beta + \frac{\sigma_{c0}}{2}\beta^2 + \frac{1}{6}\beta^3 & \alpha_c = 0 \end{cases} \quad (50)$$

STEP SIZE SELECTION AND STOPPING CRITERIA FOR ENCKE-BETA

The Encke-Beta propagator onboard Orion integrates a trajectory over the universal anomaly β from an initial value of zero at t_0 to an unknown final value β_f corresponding with t_f . Before an appropriate value of $\Delta\beta$, the fixed integration step size, can be found the value of β_f must be determined. Recall that Kepler's equation, equation 18, provides a mapping between t and β . In other words, the conic (osculating) trajectory can be used as a mechanism (via Kepler's equation) to convert between β and t and vis-a-versa. Therefore, β_f may be computed by simply solving Kepler's equation with $t = t_f$

In an earlier section Shepperd's method for solving Kepler's problem was presented. However, this method does not find β . Instead, Shepperd's method cleverly finds the values of the required Stumpff functions without computing β . While this is advantageous to Shepperd's purposes, here β must be computed. To find β apply Battin's equation 4.86 (p. 179)

$$\beta = \alpha_c \sqrt{\mu}(t - t_0) + \sigma - \sigma_{c0} \quad (51)$$

Rearranging equation 50 gives a expression for β in terms of time and the the artifacts of Shepperd's solution. Note that this equation naturally accounts for multiple revolution orbits.

$$\beta = \alpha_c \sqrt{\mu}(t - t_0) + (1 - z) [8\sigma_{c0}z + 4(1 - \alpha_c r_{c0})(1 - 2z)w] \quad (52)$$

*See Appendix C for a discussion concerning the evaluation of continued fractions.

This equation is then used to compute β_f .

The step size for the fixed step required for a total of N steps is now trivial

$$\Delta\beta = \frac{\beta_f}{N} \quad (53)$$

Since the value for $\Delta\beta$ is tied to one particular conic (osculating) trajectory, it must be recomputed at each rectification cycle. Care must be taken so that $\Delta\beta$ values for each rectification cycle are computed using the number of remaining steps for N . In this fashion, the total aggregate step count remains unchanged. Thus $\Delta\beta$ and β_f are unique for each rectification cycle, but the values of t_f and the total number of steps are constant. Figure 5 shows how this is done. The curve in Figure 5 is the solution to the differential equation 15. This curve can be viewed as a piecewise version of Keplers equation as defined in 18. Note that the slope of this curve abruptly changes* at the rectification points. This change in slope reflects the abrupt change in r_c .

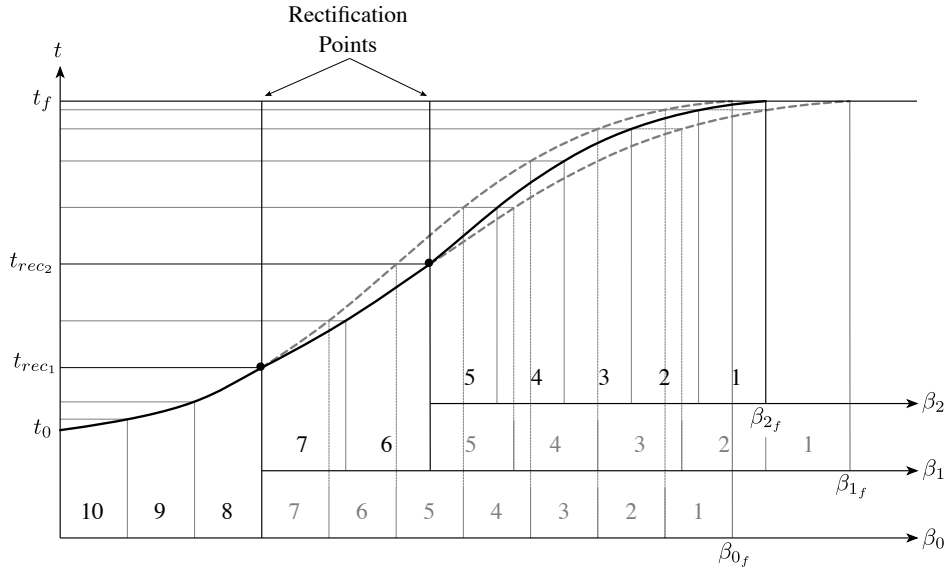


Figure 5. Step size selection and stopping criteria for Encke-Beta.

The sense of the curvature (i.e. concave-up or concave-down) for the curve in Figure 5 indicates the sign of σ_c . This is seen by noting that from equations 15 and 20 that

$$t'' = \frac{r'_c}{\sqrt{\mu}} = \frac{\sigma_c}{\sqrt{\mu}} \quad (54)$$

Thus this curve is concave-up for $\sigma_c > 0$ (i.e. positive conic flight path angles) and concave down for $\sigma_c < 0$ (i.e. negative conic flight path angles). The inflection point at the point of

*Note that this since this curve is Lipschitz continuous, the existence and uniqueness theorem from the theory of ordinary differential equations holds.

maximum slope corresponds to apoapse of the conic orbit. Similarly, the inflection point at the point of minimum slope corresponds to periapse.

Note that bounds on the change in slope at the rectification points can be written in terms of the rectification criteria. The maximum change in slope is given by

$$|\Delta t'| = \left| \left(t' \right)^+ - \left(t' \right)^- \right| \quad (55)$$

$$= \frac{|r_c^+ - r_c^-|}{\sqrt{\mu}} \quad (56)$$

From the triangle inequality it is clear that

$$|r_c^+ - r_c^-|_{max} = |r_c^- \pm \delta_{max} - r_c^-| \quad (57)$$

$$= \delta_{max} \quad (58)$$

$$\approx \left(\frac{\delta_{thresh}}{r_c^-} \right) r_c^- \quad (59)$$

Therefore a heuristic bound on the slope change at rectification points is

$$|\Delta t'|_{max} \approx \left(\frac{\delta_{thresh}}{r_c^-} \right) \left(t' \right)^- \quad (60)$$

For a small value of δ_{thresh}/r_c this change is fairly minor and the resulting change in $\Delta\beta$ is relatively small. The net effect is that $\Delta\beta$ is relatively constant across rectification cycles. That is, $\Delta\beta$ does not change in a drastic, or untenably large, way across rectification steps.

Furthermore, since the remaining time, $t_f - t_{rec}$, is a constant across any rectification cycle, it is easily observed that any change in $\Delta\beta$ results in an even smaller change to the times of the individual steps. In other words, the distribution of the remaining steps in time is nearly constant across rectification cycles. This phenomena is represented by the small dotted lines in Figure 5. However, these times can drift gradually as the perturbations aggregate over an extended period of time.

NUMERICAL COMPARISON OF ENCKE-BETA AND ENCKE-TIME

The central role of r_c in the Sundman transformation suggests a strong dependence on eccentricity of the orbit. In the case of a circular orbit ($e = 0$) the value of r_c is constant and the Sundman transformation becomes a simple scaling of the independent variables. Thus for a circular orbit, the expectation is that Encke-Beta and Encke-Time will be nearly identical in terms of performance. As the eccentricity increases r_c begins to vary and the Sundman transformation causes fixed steps in β to become modulated steps in time. This effect is shown in Figure 5. Note that for a circular orbit the curve in figure 5 would be straight. As the eccentricity of the orbit increases the curvature of the curve in figure 5 increases. This increase results in the desired step size modulation.

To observe this effect, and understand its consequences, orbits of varying eccentricity are used to compare the fixed step performance improvements available by using Encke-Beta rather than Encke-Time. The same orbits were also integrated using Cowell's method. Each orbit propagation begins at periapse and lasts for 30 hours. The magnitude of the velocity at periapse is adjusted to change the eccentricity of each orbit. The relative error from some of these experiments is shown in Figure 7a

The error plots in Figure 7a show the characteristic features expected for Runge-Kutta class integration schemes. For step sizes that are too large (too few steps over a fixed interval) the dominant error is approximation error. Approximation error is due the numerical integration scheme failing to produce an accurate approximation over such a large step size. For excessively small step sizes (too many steps over a finite interval) the error is dominated by the effects of finite machine precision. Finite precision error is characterized by further reductions in step size producing little or no reduction in error. Appropriate steps which are neither too large nor too small exist in the goldilocks region. On a log-log plot of error vs number of steps (or step size), such as figure 7a, the goldilocks zone is easily identified by a straight line with a slope of approximately the order of the integrator. The straight line is a result of the integration error being characterized by

$$\text{error} = Kh^n \quad (61)$$

where K is a constant, h is the step size, and n is the order of integration scheme. Even when the truth is not known this characteristic straight line can be identified by

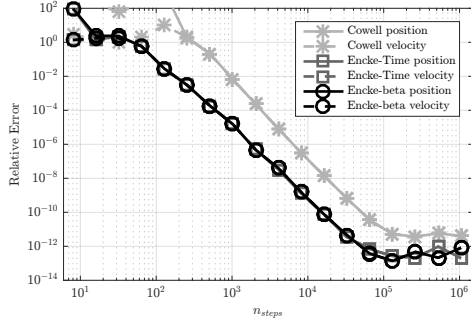
$$\log_k \frac{|x(h) - x(kh)|}{|x(\frac{h}{k}) - x(h)|} \approx n \quad (62)$$

where k is a real number, and $x(h)$ is the result of integrating the equations with a step size of h . [22]

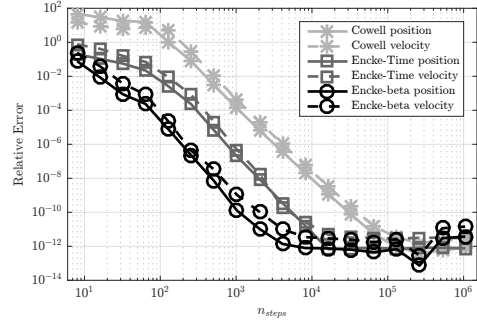
It is useful to examine the difference between Encke-Time and Encke-Beta from two different perspectives. First, compare the accuracy available for both methods given a fixed amount of steps. This corresponds to the vertical distance separating the error lines from Encke-Time and Encke-Beta. Figure 7a shows the factor representing the accuracy improvement possible by moving from Encke-Time to Encke-Beta while keeping the number of integration steps fixed. As expected, this factor is near unity (no improvement) for circular orbits but dramatically increases for highly elliptic orbits. For hyperbolic orbits the improvement plateaus. For Molniya orbits the factor is near 1000. For example, this means that for orbits with $e \approx 0.74$ if the Encke-Time routine is accurate on the order of kilometers, the Encke-Beta routine is expected to be accurate on the order of meters. For parabolic and hyperbolic orbits the factor increases to more than 100000. This corresponds to a difference between kilometers and centimeters.

Figure 7b view the performance gains from the second perspective. Here the factor represents the difference in the number of steps (or function evaluations) required to achieve a specified accuracy. For example, for a Molniya orbit Encke-Time will require approximately $4\times$ the number of steps that Encke-Beta requires to achieve the same level of accuracy. For parabolic and Hyperbolic orbits Encke-Time requires more than $10\times$ the number of steps that Encke-Beta requires to deliver the same level of accuracy.

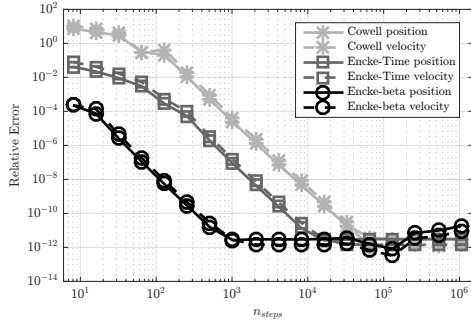
This analysis shows that Encke-Beta integration offers a significant performance improvement when comparing Encke-Time for fixed step size integration of highly elliptic orbits. For these highly eccentric orbits the performance gain between Encke-Time and Encke-Beta is even greater than the performance gain between Cowell and Encke-Time. Although the performance gains are more modest for orbits with low eccentricity they are still significant. No cases were found where Encke-Time offers superior performance when compared to Encke-Beta.



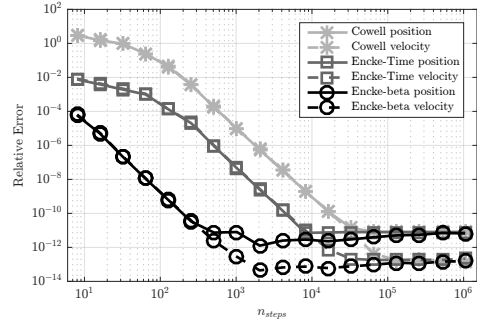
(a) An orbit with $e = 0.00$ (Circular).



(b) An orbit with $e = 0.74$ (Molniya).

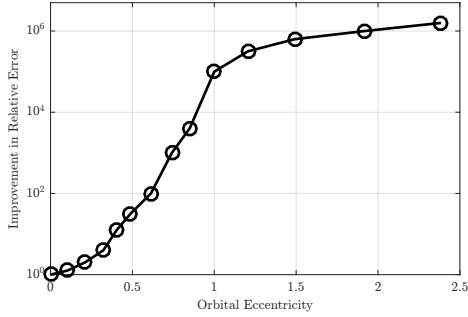


(c) An orbit with $e = 1.00$ (Parabolic).

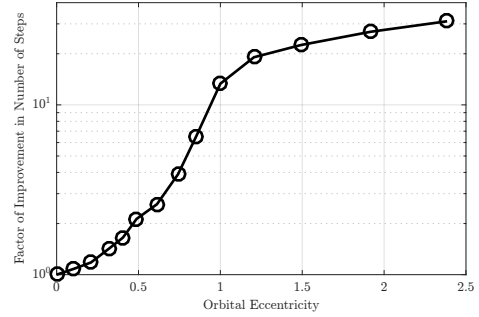


(d) An orbit with $e = 1.49$ (Hyperbolic).

Figure 6. Difference in relative error between Cowell, Encke-Time, and Encke-Beta for orbits of distinct eccentricity.



(a) The factor for the available error reduction by using Encke-Beta rather than Encke-Time for a fixed step size.



(b) The factor for the number of steps reduction available by using Encke-Beta rather than Encke-Time to meet a fixed accuracy requirement.

Figure 7. Factors showing the efficiency gains possible by moving from Encke-Time to Encke-Beta.

CONCLUSIONS

The Encke-Beta method provides Orion targeting and guidance with a trajectory prediction scheme that is valid for circular, elliptical, parabolic, and hyperbolic orbits. The use of a pre-determined number of beta steps during the integration permits the trajectory prediction to complete within a specified guidance cycle duration. The use of an independent variable (beta) that is a function of eccentric anomaly results in beta steps of smaller duration in terms of time at and near perigee, when the velocity vector changes rapidly in terms of direction and magnitude. This results in the higher accuracy of Encke-Beta as opposed to Encke-Time as eccentricity increases.

ACKNOWLEDGMENTS

James J. McMahon provided information on past implementations of the Encke-Beta algorithm. [23] While employed by TRW in Houston he worked on the Encke-Beta and Encke-Time software used in Mission Control for the Apollo and Space Shuttle Programs. Apollo veteran A. David Long provided the authors his 1970 analysis of the Encke-Time trajectory predictor in the Lunar Module flight software.

APPENDIX A: TRAJECTORY PREDICTORS USED BY PREVIOUS NASA HUMAN FLIGHT PROGRAMS

Previous NASA human space flight programs have used methods for state vector prediction.* This appendix focuses mostly on the use of Encke methods.

In addition to the use of trajectory prediction methods in on-board flight computers, ground personnel for human flight missions and non-human flight missions (satellites, interplanetary probes, etc.) use trajectory prediction algorithms as part of the orbital determination process, for burn planning, and other mission planning applications. Prediction algorithms are used before a flight, during the flight, and as part of post-flight trajectory reconstruction.

Mission Control Use of Encke's Method

During Apollo Encke's method with both time and beta options was used in the Free Flight Numerical Integrator (FFNI) in Mission Control at the Manned Spacecraft Center in Houston. The Encke-Beta method was preferred over Encke-Time since Encke-Beta provided the same numerical accuracy as the Cowell and Encke-Time methods but required less execution time. The Apollo FFNI was later adapted for the Skylab Program (1973-1974), and the Skylab version served as the basis for the FFNI algorithm used during the Apollo-Soyuz Test Project (ASTP) in 1975.

In 1976 the ASTP version of the FFNI was adapted for use by the Space Shuttle Program. The Encke-Time and Encke-Beta methods for the Space Shuttle were implemented in FORTRAN IV and Assembly Language in the Mission Operations Computer (MOC), and supported flights from 1981 through 2002. Space Shuttle Encke was later re-written in C as part of the Trajectory-Off-the-MOC Project and resided on the new Trajectory Server[†]. It was used in Mission Control from 2002 until the end of the Space Shuttle Program in 2011.

*NASA Goddard used Encke-Beta for interplanetary trajectory prediction in the 1960s. [24]

[†]The Trajectory Server first supported the on-orbit and entry phases of STS-110 (April 2002). It then supported all flight phases of STS-111 (June 2002) with the MOC as a backup. The MOC was finally powered down on August 12, 2002.

Apollo On-Board Use of Encke's Method

An Encke-Time algorithm called the Coasting Integration Routine was used in the Apollo Command Service Module (CSM) computer software (called COLOSSUS) and the Lunar Module (LM) primary computer software (called LUMINARY). [3] Nyström's integration method was used, providing fourth order accuracy with three derivative evaluations per time step.

The Coasting Integration Routine was used by LUMINARY to predict state vectors for both the LM and CSM. State vector prediction was used during various activities, including rendezvous radar updates both in lunar orbit or while on the lunar surface, predicting the vehicle state vector to the burn time-of-ignition, rendezvous burn targeting, and predicting the state vectors to the current time in support of other software tasks. Use of the LUMINARY Coasting Integration Routine was restricted to Earth or lunar orbit. It did not include third body perturbations. The only disturbing acceleration was non-spherical gravitational perturbations of the primary body.

The COLOSSUS Coasting Integration Routine also predicted both CSM and LM state vectors. The Coasting Integration Routine was used during orbit navigation (Earth and lunar), rendezvous navigation, and cislunar-midcourse navigation. The routine was used in conjunction with the Kalman filtering process, when the following measurements were taken and filtered: on-orbit navigation (landmark tracking with the scanning telescope or sextant), cislunar navigation (star-landmark or star-horizon measurements with the sextant), and rendezvous navigation (sextant angles and VHF ranging). In Earth or lunar orbit the Coasting Integration Routine did not include third body perturbations. The only disturbing acceleration was non-spherical gravitational perturbations of the primary body. During cis-lunar flight acceleration included the primary body (Earth or Moon), and disturbing acceleration from two secondary bodies (the Sun and either the Moon or Earth).

Space Shuttle On-Board Precision Predictor

The trade study for the Space Shuttle on-board precision predictor was conducted circa 1976 within the Mathematical Physics Branch (FM8) of the Mission Planning and Analysis Division of the NASA/Johnson Space Center. The candidates were Pines' Method and two variations of the KS method. [25, 26, 27] Pines' required fewer integration variables, less memory, and was faster than the Kustaanheimo-Stiefel methods. However, the KS methods were more stable for long predictions. Pines' Method was chosen and was used to: 1) predict uplinked state vectors to the current time or the Time of Ignition of a burn, 2) perform orbiter and target spacecraft predictions during Lambert Targeting of rendezvous burns, and 3) predict state vectors during rendezvous when performing relative navigation. Pines' Method could only support elliptical and circular orbits.

APPENDIX B: THE STUMPF FUNCTIONS

In this section, analytic expressions are given for the Stumpff functions required to solve Kepler's problem. Analytic expressions for the first six Stumpff functions are given in table 1.

Shepperd's w function is then given by

$$w = \begin{cases} \frac{\tan(\sqrt{\alpha}\frac{\beta}{4})}{\sqrt{\alpha}} & \alpha > 0 \\ \frac{\beta}{4} & \alpha = 0 \\ \frac{\tanh(\sqrt{-\alpha}\frac{\beta}{4})}{\sqrt{-\alpha}} & \alpha < 0 \end{cases} \quad (63)$$

Table 1. Analytical expressions for the Stumpff functions required to solve Kepler's problem and compute the associated state transition matrix.

Stumpff Function	$\alpha > 0$ (Elliptic)	$\alpha = 0$ (Parabolic)	$\alpha < 0$ (Hyperbolic)	Figure #
$U_0(\beta; \alpha)$	$\cos(\sqrt{\alpha}\beta)$	1	$\cosh(\sqrt{-\alpha}\beta)$	Figure 8a
$U_1(\beta; \alpha)$	$\frac{\sin(\sqrt{\alpha}\beta)}{\sqrt{\alpha}}$	β	$\frac{\sinh(\sqrt{-\alpha}\beta)}{\sqrt{-\alpha}}$	Figure 8b
$U_2(\beta; \alpha)$	$\frac{1 - \cos(\sqrt{\alpha}\beta)}{\alpha}$	$\frac{\beta^2}{2!}$	$\frac{1 - \cosh(\sqrt{-\alpha}\beta)}{\alpha}$	Figure 8c
$U_3(\beta; \alpha)$	$\frac{\sqrt{\alpha}\beta - \sin(\sqrt{\alpha}\beta)}{\alpha\sqrt{\alpha}}$	$\frac{\beta^3}{3!}$	$\frac{\sqrt{-\alpha}\beta - \sinh(\sqrt{-\alpha}\beta)}{\alpha\sqrt{-\alpha}}$	Figure 8d
$U_4(\beta; \alpha)$	$\frac{\frac{(\sqrt{\alpha}\beta)^2}{2!} - 1 + \cos(\sqrt{\alpha}\beta)}{\alpha^2}$	$\frac{\beta^4}{4!}$	$\frac{\frac{-(\sqrt{-\alpha}\beta)^2}{2!} - 1 + \cosh(\sqrt{-\alpha}\beta)}{\alpha^2}$	Figure 8e
$U_5(\beta; \alpha)$	$\frac{\frac{(\sqrt{\alpha}\beta)^3}{3!} - \sqrt{\alpha}\beta + \sin(\sqrt{\alpha}\beta)}{\alpha^2\sqrt{\alpha}}$	$\frac{\beta^5}{5!}$	$\frac{\frac{-(\sqrt{-\alpha}\beta)^3}{3!} - \sqrt{-\alpha}\beta + \sinh(\sqrt{-\alpha}\beta)}{\alpha^2\sqrt{-\alpha}}$	Figure 8f
$U_6(\beta; \alpha)$	$\frac{\frac{(\sqrt{\alpha}\beta)^4}{4!} - \frac{(\sqrt{\alpha}\beta)^2}{2!} + 1 - \cos(\sqrt{\alpha}\beta)}{\alpha^3}$	$\frac{\beta^6}{6!}$	$\frac{\frac{(\sqrt{-\alpha}\beta)^4}{4!} - \frac{(\sqrt{-\alpha}\beta)^2}{2!} + 1 - \cosh(\sqrt{-\alpha}\beta)}{\alpha^3}$	Figure 8g

This equation is shown in Figure 9. Equations 36-38 show how to compute $U_0(\beta; \alpha)$ through $U_2(\beta; \alpha)$.

Evaluation of the higher order Stumpff functions can be accomplished by using the hypergeometric function $G(5, 0, 5/2, z)$. It can be shown hypergeometric function $G(5, 0, 5/2, z)$ has the analytic expression

$$G\left(5, 0, \frac{5}{2}, z\right) = \begin{cases} \frac{5}{128(1-z)^2 z^2} \left[3 \frac{\sin^{-1}(\sqrt{z})}{\sqrt{z(1-z)}} + (1-2z) \left(2(1-2z)^2 - 5 \right) \right] & z > 0 \\ 1 & z = 0 \\ \frac{5}{128(1-z)^2 z^2} \left[3 \frac{\sinh^{-1}(\sqrt{-z})}{\sqrt{z(z-1)}} + (1-2z) \left(2(1-2z)^2 - 5 \right) \right] & z < 0 \end{cases} \quad (64)$$

This expression is singular at $z = 1$ but this is not problematic since $z \in (-\infty, 1/2]$. $G(5, 0, 5/2, z)$ is shown in Figure 10. Note the following two easily obtained but nontrivial special values $G(5, 0, 5/2, \frac{1}{2}) = \frac{15\pi}{16}$ and $G(5, 0, 5/2, \frac{1}{4}) = \frac{5}{54} (8\sqrt{3}\pi - 27)$. Also note that

$$\lim_{z \rightarrow -\infty} G\left(5, 0, \frac{5}{2}, z\right) = 0 \quad (65)$$

Since $G(5, 0, 5/2, z)$ is monotonically increasing (for $z \in (-\infty, 1/2]$), it follows directly that $G(5, 0, 5/2, z) \in (0, \frac{15\pi}{16}]$

To find the higher order Stumpff functions compute the parameter

$$Q = \frac{4}{3} \left[1 - 2q + q(1-q) \frac{16}{5} G\left(5, 0, \frac{5}{2}, z\right) \right] \quad (66)$$

The higher order Stumpff functions are

$$U_3(\beta; \alpha) = \left[\frac{1}{2} Q U_1^2(\beta; \alpha) - U_2(\beta; \alpha) \right] U_1(\beta; \alpha) \quad (67)$$

$$U_4(\beta; \alpha) = U_1(\beta; \alpha) U_3(\beta; \alpha) - \frac{1}{2} [U_2^2(\beta; \alpha) - \alpha U_3^2(\beta; \alpha)] \quad (68)$$

$$U_5(\beta; \alpha) = \frac{1}{3} \left[\frac{16}{5} G\left(5, 0, \frac{5}{2}, z\right) (2(1-z)w)^5 - U_2(\beta; \alpha) U_3(\beta; \alpha) - \beta U_4(\beta; \alpha) \right] \quad (69)$$

$$U_6(\beta; \alpha) = \frac{1}{2} [U_3^2(\beta; \alpha) - U_2(\beta; \alpha) U_4(\beta; \alpha)] - \frac{1}{4} \beta^2 U_4(\beta; \alpha) + \beta U_5(\beta; \alpha) \quad (70)$$

APPENDIX C: EVALUATION OF THE CONTINUED FRACTIONS

This section contains methods for rapidly evaluating the continued fractions in equations 42 and 35. Both continued fractions are evaluated using the top-down method presented by Gautschi in 1967. [28] Battin and Shepperd both used this method for top-down continued fraction evaluation. [3, 17]

Gautschi's algorithm evaluates the continued fraction

$$s = \frac{n_0/d_0}{1 - \frac{\frac{x n_1/d_1}{1 - \frac{x n_2/d_2}{1 - \frac{x n_3/d_3}{\ddots}}}} \quad (71)$$

in a top-down fashion by using the following iteration scheme.

$$p_k = \frac{d_k}{d_k - n_k p_{k-1} x} \quad \text{with} \quad p_0 = 1 \quad (72)$$

$$u_k = u_{k-1} (p_k - 1) \quad \text{with} \quad u_0 = n_0/d_0 \quad (73)$$

$$s_k = s_{k-1} + u_k \quad \text{with} \quad s_0 = n_0/d_0 \quad (74)$$

stopping once s_k converges. It will now be shown how both equation 42 and 35 can be evaluated in this manner.

Equation 42 is a special case of the Gaussian continued fraction $G(A, B, C, x)$ where for $k \in \{1, 2, 3, \dots\}$

$$\frac{n_{k_{\text{even}}}}{d_{k_{\text{even}}}} = \frac{(k+2B)(k+2C-2A)}{4(k+C-1)(k+C)} \quad \frac{n_{k_{\text{odd}}}}{d_{k_{\text{odd}}}} = \frac{(k+2A-1)(k+2C-2B-1)}{4(k+C-1)(k+C)} \quad (75)$$

and $n_0/d_0 = 1$. Note the denominator d_k has the same form for both odd and even values of k . It is not difficult to show that this can be written

$$d_k = 4C^2 + 4(2k-1)C + 4k(k-1) \quad (76)$$

which leads directly to the additive recursion relation

$$d_k = d_{k+1} + 4 \cdot \underbrace{2(C+k-1)}_{l_k} \quad \text{with} \quad d_0 = 4C(C-1) \quad (77)$$

where

$$l_k = l_{k+1} + 2 \quad \text{with} \quad l_0 = 4C(C-1) \quad (78)$$

An additive recursion for the numerator n_k requires dealing with odd-to-even and even-to-odd transitions distinctly. Nevertheless, it will be shown that a simple additive recursion exists. Careful examination of the expanded relations for the numerator

$$n_{k_{even}} = 4B(C - A) + 2k(-A + B + C) + k^2 \quad (79)$$

$$n_{k_{odd}} = 4A(C - B) + 2(k - 1)(A - B + C) + (k - 1)^2 \quad (80)$$

leads to the following additive recursions

$$n_{k_{even}} = n_{(k-1)_{odd}} + 4(C + k - 1)(B - A + 1) \quad (81)$$

$$n_{k_{odd}} = n_{(k-1)_{even}} + 4(C + k - 1)(A - B) \quad (82)$$

These two equations can be written as a single equation

$$n_k = n_{k-1} + \overbrace{2(C + k - 1)}^{l_k} \left[\overbrace{1 \pm (1 - 2A + 2B)}^{K_k} \right] \quad \text{with} \quad n_0 = 4B(C - A) \quad (83)$$

where the “+” corresponds with equation 81 and the “−” corresponds with equation 82. These additive recursions allow for rapid evaluation of an arbitrary Gaussian continued fraction using Gautschi’s method. For the special case of equation 42, $G(5, 0, 5/2, x)$, further simplification is possible resulting in

$$K_k = -K_{k-1} \quad \text{with} \quad K_0 = -9 \quad (84)$$

$$l_k = l_{k-1} + 2 \quad \text{with} \quad l_0 = 3 \quad (85)$$

$$n_k = n_{k-1} + (1 + K_k) l_k \quad \text{with} \quad n_0 = 0 \quad (86)$$

$$d_k = d_{k-1} + 4l_k \quad \text{with} \quad d_0 = 15 \quad (87)$$

Before Gautschi’s method can be applied to equation 35 an equivalent continued fraction must be found.

$$w = \frac{\frac{1}{4}\beta}{1 - \frac{\alpha_c(\frac{1}{4}\beta)^2}{3 - \frac{\alpha_c(\frac{1}{4}\beta)^2}{5 - \frac{\alpha_c(\frac{1}{4}\beta)^2}{7 - \ddots}}}} = \frac{\frac{1}{4}\beta}{1 - \frac{(1/3)\alpha_c(\frac{1}{4}\beta)^2}{1 - \frac{(1/15)\alpha_c(\frac{1}{4}\beta)^2}{1 - \frac{(1/35)\alpha_c(\frac{1}{4}\beta)^2}{1 - \ddots}}}} \quad (88)$$

Now the application of Gautschi’s method is readily apparent

$$x = \alpha_c \left(\frac{1}{4}\beta\right)^2 \quad (89)$$

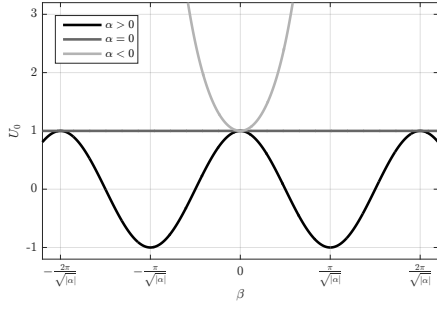
$$l_k = l_{k-1} + 2 \quad \text{with} \quad l_0 = 1 \quad (90)$$

$$n_k = 1 \quad (91)$$

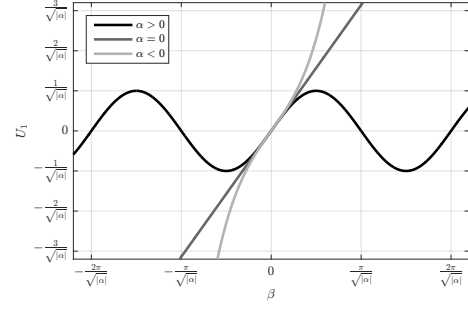
$$d_k = l_k l_{k-1} \quad \text{with} \quad d_0 = 1 \quad (92)$$

After convergence the to the final value, u_{final} , the w can be easily computed by

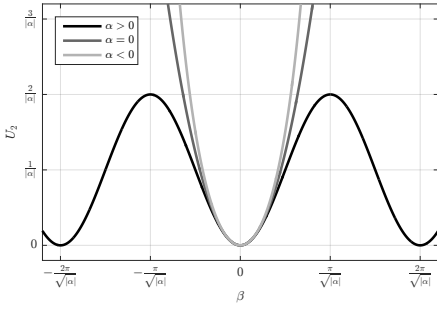
$$w = \frac{\beta}{4} u_{final} \quad (93)$$



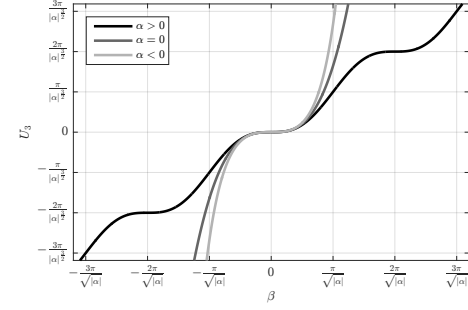
(a) The Stumpff function $U_0(\beta; \alpha)$.



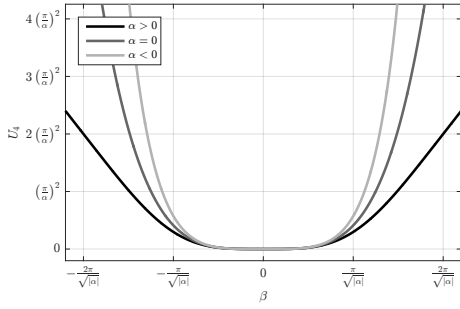
(b) The Stumpff function $U_1(\beta; \alpha)$.



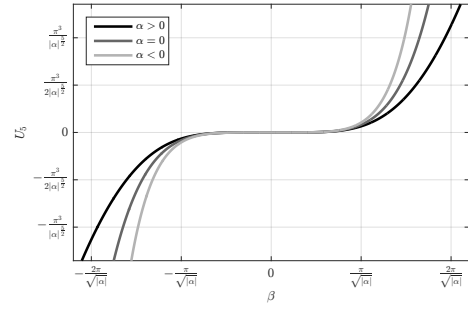
(c) The Stumpff function $U_2(\beta; \alpha)$.



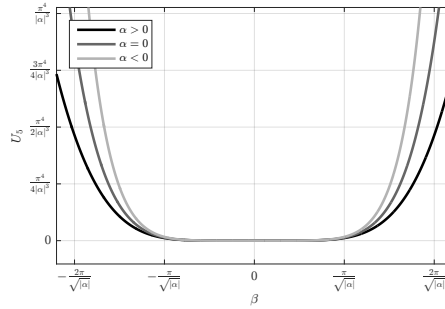
(d) The Stumpff function $U_3(\beta; \alpha)$.



(e) The Stumpff function $U_4(\beta; \alpha)$.



(f) The Stumpff function $U_5(\beta; \alpha)$.



(g) The Stumpff function $U_6(\beta; \alpha)$.

Figure 8. Stumpff functions.

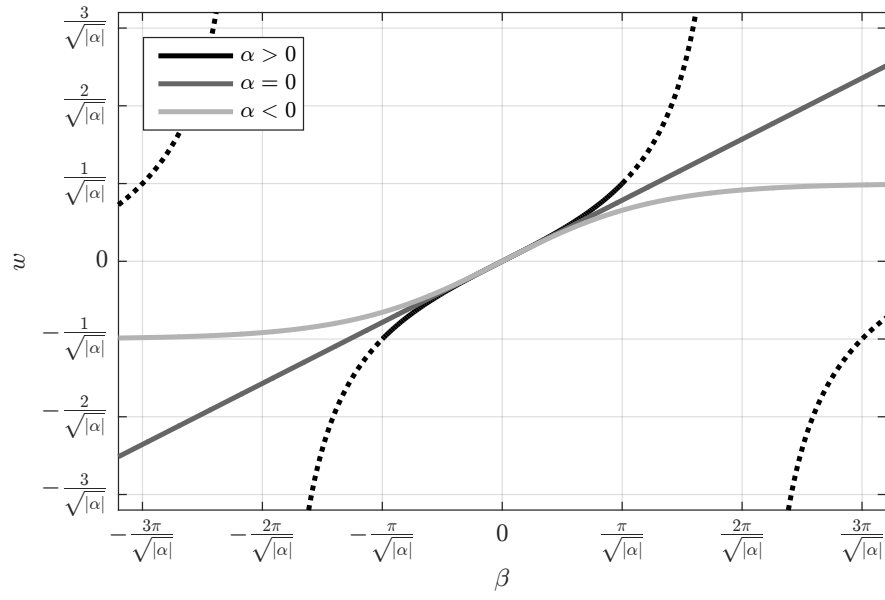


Figure 9. Shepperd's w function.

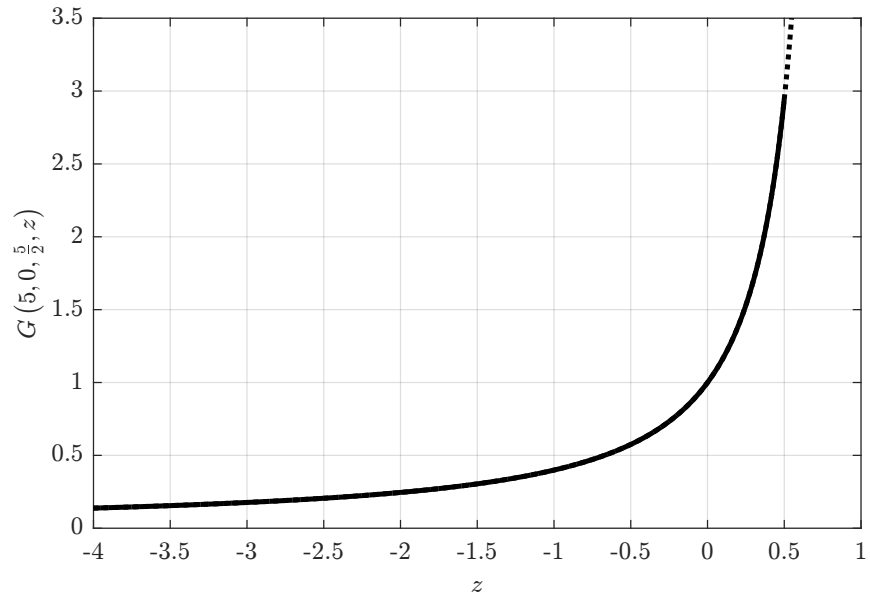


Figure 10. The hypergeometric function $G\left(5, 0, \frac{5}{2}, z\right)$.

REFERENCES

- [1] J. F. Encke, “Über eine neue Methode der Berechnung der Planetenstörungen,” *Astronomische Nachrichten*, Vol. 33, February 1852, pp. 377–398, 10.1002/asna.18520332602.
- [2] R. R. Bate, D. D. Mueller, and J. E. White, *Fundamentals of Astrodynamics*, pp. 390–396. New York, NY: Dover Publications, 1971.
- [3] R. H. Battin, *An Introduction to the Mathematics and Methods of Astrodynamics*, pp. 174–179, 447–450, 760–761. AIAA Education Series, Reston, Virginia: American Institute of Aeronautics and Astronautics, revised ed., 1999.
- [4] P. H. Cowell and A. C. D. Crommelin, “The Orbit of Jupiter’s Eighth Satellite,” *Notices of the Royal Astronomical Society of London*, Vol. 68, 1908, pp. 576–581.
- [5] J. J. McMahon, *Fundamentals of Interplanetary Trajectory Simulation*, pp. 16–19. Self Published, 2012.
- [6] R. H. Battin, *Astronautical Guidance*, pp. 188–190, 216, 218. New York, NY: McGraw-Hill, 1964.
- [7] R. M. L. Baker, *Astrodynamics, Applications and Advanced Topics*, pp. 227–242. New York, NY: Academic Press, 1967.
- [8] S. K. Scarritt, T. J. Fill, and S. B. Robinson, “Advances in Orion’s On-Orbit Guidance and Targeting System Architecture,” *AAS Guidance and Control Conference*, No. AAS 15-106, Breckenridge, CO, January-February 2015.
- [9] S. K. Scarritt, B. G. Marchand, A. J. Brown, W. H. Tracy, and M. W. Weeks, “Finite-Burn Linear Targeting Algorithm for Autonomous Path Planning and Guidance,” *Journal of Guidance, Control, and Dynamics*, Vol. 35, September-October 2012, pp. 1605–1615, 10.2514/1.54249.
- [10] S. W. Thrasher and T. J. Fill, “Orion’s Exoatmospheric Burn Guidance Architecture and Algorithm,” *AIAA Guidance, Navigation, and Control Conference*, No. AIAA 2011-6262, Portland, OR, American Institute of Aeronautics and Astronautics, 2011, 10.2514/6.2011-6262.
- [11] R. L. McHenry, T. J. Brand, A. D. Long, B. F. Cockrell, and J. R. Thidodeau, III, “Space Shuttle Ascent Guidance, Navigation, and Control,” *The Journal of the Astronautical Sciences*, Vol. 27, January-March 1979, pp. 1–38.
- [12] V. R. Bond and M. C. Allman, *Modern Astrodynamics: Fundamental and Perturbation Methods*, pp. 144–177. Princeton University Press, 1996.
- [13] R. G. Gottlieb, “Variation of Parameters Methods Embedding the Jacobi Integral Resulting from Energy Parameter Selection,” *AAS/AIAA Space Flight Mechanics Meeting*, No. AAS 99-202, Breckenridge, CO, American Astronautical Society/American Institute of Aeronautics and Astronautics, February 1999.
- [14] A. E. Roy, *Orbital Motion*, pp. 200–203. New York, NY: John Wiley & Sons, 1978.
- [15] J. M. A. Danby, *Fundamentals of Celestial Mechanics*, pp. 117–118. Willmann-Bell, Inc., second ed., 1996.
- [16] K. J. N. Stumpff, “Neue Formeln und Hilfstafeln zur Ephemeridenrechnung,” *Astronomische Nachrichten*, Vol. 275, No. 7-9, 1947, pp. 108–128, 10.1002/asna.19472750703.
- [17] S. W. Shepperd, “Universal Keplerian State Transition Matrix,” *Celestial Mechanics and Dynamical Astronomy*, Vol. 35, February 1985, pp. 129–144, 10.1007/BF01227666.
- [18] S. Herrick, “Universal Variables,” *The Astronomical Journal*, Vol. 70, May 1965, pp. 309–316, 10.1086/109728.
- [19] S. Herrick, “Errata: Universal variables,” *The Astronomical Journal*, Vol. 70, August 1965, p. 447, 10.1086/109763.
- [20] W. M. Lear, “Nystrom Integrators,” Tech. Rep. JSC-24229, NASA - Lyndon B. Johnson Space Center, March 1990.
- [21] W. M. Lear, “Kalman Filtering Techniques,” Tech. Rep. JSC-20688, NASA - Lyndon B. Johnson Space Center, Houston, Texas, September 1985.
- [22] J. H. Hubbard and B. H. West, *Differential Equations: A Dynamical Systems Approach*, Vol. 5 of *Texts in Applied Mathematics*. Springer, 1991.
- [23] D. R. Adamo, “A Precision Orbit Predictor Optimized for Complex Trajectory Operations,” *AAS/AIAA Astrodynamics Specialists Conference*, No. AAS 03-665, August 2003.
- [24] L. Lefton and F. Whitlock, “Interplanetary Trajectory Method (ITEM) Update 12C Program Manual,” Technical Memorandum NASA TM X-55841, NASA - Goddard Space Flight Center, June 1967.
- [25] E. L. Stiefel and G. Scheifele, *Linear and Regular Celestial Mechanics: Perturbed Two-body Motion, Numerical Methods, Canonical Theory*. Berlin: Springer-Verlag, 1971.

- [26] S. Pines, "Variation of Parameters for Elliptic and Near Circular Orbits," *Astronomical Journal*, Vol. 66, February 1961, pp. 5–7, 10.1086/108348.
- [27] S. Pines, "Initial Cartesian Coordinates for Rapid Precision Orbit Prediction," technical report, NASA - Lyndon B. Johnson Space Center, 1975. in Research in Navigation and Optimization for Space Trajectories, October 1, 1975 through September 30, 1979, NASA CR-160420, 1979.
- [28] W. Gautschi, "Computational Aspects of Three-Term Recurrence Relations," *SIAM Review*, Vol. 9, January 1967, pp. 24–82, 10.1137/1009002.