STATE TRANSITION TENSOR MODELS FOR THE UNCERTAINTY PROPAGATION OF THE TWO-BODY PROBLEM

Tarek A. Elgohary[*] and James D. Turner[†]

Several methods exist for integrating the Keplerian Motion of two gravitationally interacting bodies, even when gravitational perturbation terms are included. The challenge is that the equations of motion become very stiff when the perturbation terms are included, which forces the use of small time steps, higher-order methods, or extended precision calculations. Recently, Turner and Elgohary have shown that by introducing two scalar Lagrange-like invariants that it is possible to integrate the two-body and two-body plus *J2* perturbation term using a recursive formulation for developing an analytic continuation-based power series that overcomes the limitations of standard integration methods. Numerical comparisons with RK12(10), and other state of the art integration methods indicate performance improvements of ~ 70X, while maintaining ~mm accuracy for the orbit predictions. Extensions for *J3* through *J6* are currently under development. With accurate trajectories available, the next important theoretical development becomes extending the series-based solution for the state transition matrices (STM) for both the two-body and two-body plus *J2* perturbation. STMs are useful for many celestial mechanics optimization calculations. Second and third order STM models are developed to support uncertainty propagation investigations. The application of scalar Lagrange-like invariants generates highly efficient state trajectory, STM, and higher-order STMs models. The proposed mathematical models are expected to be broadly useful for celestial mechanic applications for optimization, uncertainty propagation, and nonlinear estimation theory.

INTRODUCTION

The development of STM models requires an accurate trajectory prediction for building the nonlinear STM ordinary differential equations. For the unperturbed two-body problem the classical F&G Lagrangian coefficients provide a mapping of the initial position and velocity into current time values. Majji, Junkins, and Turner,[1] based on the F&G model, have developed first-through fifth-order STM models. Unfortunately, the F&G STM-based model is not well suited for being extended for handling J2 and higher-order gravity perturbation effects. This work presents an analytic series-based approach that overcomes the problems encountered for applying the F&G STM for gravity perturbations.

[*] Graduate Research Assistant, Aerospace Engineering, Texas A&M University, College Station, Texas, t.gohary@gmail.com
[†] Research Professor, Aerospace Engineering, Texas A&M University, College Station, Texas, turner@aero.tamu.edu, Fellow AAS, Associate Fellow AIAA

Continuation Methods for the Two-Body Problem

A problem of long-standing for the two-body problem has been the difficulty in developing high-order Taylor series expansions, even though the governing equations are well-defined. Lagrange introduced three algebraic invariants that permit the unperturbed two-body trajectory to be expanded in a power series model.[2] However, Lagrange's analysis objective was not to develop arbitrary order time derivative models for the trajectories. Later work by Battin and others extended this approach and developed recursive algorithms for propagating the Lagrange invariants.[3, 6-8] The Lagrange invariants, however, have not been extended for handling gravity perturbations. This work handles both unperturbed and perturbed gravity external loads. All of the new algorithms build on Turner and Elgohary two scalar Lagrange-like invariants (i.e., $f = r \cdot r$ and $g = f^{-n/2}$).[4] This approach is highly efficient because the differential equations for both f and g lead to *bilinear equations*, with no denominators, which are ideally suited for generating recursively-based arbitrary-order time derivatives by invoking the use of *Leibniz product rule*. Ordinarily, without introducing the f and g variables, the calculations for *n-th* order time derivatives of the vector-valued function $(r\ r)^{-3/2}$ rapidly becomes unwieldy and very complicated for coding and validating the model. Indeed, it is likely the complexity of these calculations provided Lagrange with amply motivation for seeking out the development of his algebraic invariant approach. Using the f and g model, there is no theoretical limit to the number of series terms that can be retained in the approximation. Numerical experiments, however, demonstrate that depending on the propagation time step size and the orbit eccentricity that 8 to 15 series terms yield operationally useful approximations for missions operating in the LEO-GEO Theater of operations. Extended precision experiments, including Q-level and 200-500 digit calculations, support the conclusion that the integration time step for the trajectory analysis is limited by a fundamental *radius of convergence issue* for the series approximation.[5] On-going research efforts are exploring the *radius of convergence issue* and its impact on building operationally useful tools for supporting space trajectory applications. Of course, f and g are just scalar kinematic variables. Trajectory models require that f and g are linked with the two-body problem acceleration, defined by $r = -\mu r g$, where $r = \begin{bmatrix} x & y & z \end{bmatrix}^T$ denotes the inertial relative coordinate vector that locates an object relative to the Earth, $\mu = 398601.2$ km^2/sec^3 is the gravitational constant. It is important to note that $r = -\mu r g$ consists of a *bilinear term* involving the product of the position vector and the g scalar Lagrange-like invariant. Like the f and g variables themselves, the higher time derivatives for $r = -\mu r g$ are generated by invoking *Leibnitz product rule*. A complete series solution is generated by recursively linking the Leibnitz product rules for f, g, and r, and recovering one derivative order for each sweep of the three Leibnitz product rules: the approach is both efficient and accurate. Calculations begin by assuming that initial values are available for r, r, r, higher-order time derivatives are developed recursively linking Leibniz product rule for f, g, and r to recursively develop $\{f, g, r\}, \{f, g, r\}$, . After computing a pre-set number of terms recursively, one develops the trajectory and trajectory time derivatives as the analytic continuation power series approximations,

$$r(t+h) = r(t) + r(t)h + r(t)h^2 / 2! + r(t)h^3 / 3! +$$

$$(1)$$

$$r(t+h) = r(t) + r(t)h + r(t)h^2 / 2! +$$

Only the position and velocity vectors must be analytically continued. This work extends these results to include J_2 disturbance acceleration terms for the trajectory propagation, as well as developing analytic continuation power series for the STM. Numerical results are presented that compare the solution accuracy and integration time required by *ODE45* and *RKN1210*, and the analytic power series methods developed in this work for two-body plus J_2 perturbation terms.

ANALYTIC CONTINUATION SOLUTION FOR THE PERTURBED TWO-BODY PROBLEM

The trajectory model for the perturbed relative two-body problem is addressed with the following equation of motion,

$$r = -\frac{\mu}{r^3} r + a_d \tag{2}$$

The disturbance acceleration term $\mathbf{a}_d$, referring to the J_2 effects, that arising from the Earth's oblateness is defined as,

$$\mathbf{a}_{J_2} = -\frac{3}{2} J_2 \left(\frac{\mu}{r^2}\right)\left(\frac{r_{eq}}{r}\right)^2 \left\{\left(1-5\left(\frac{z}{r}\right)^2\right)\frac{x}{r}, \quad \left(1-5\left(\frac{z}{r}\right)^2\right)\frac{y}{r}, \quad \left(3-5\left(\frac{z}{r}\right)^2\right)\frac{z}{r}\right\} \tag{3}$$

where, $\mu = 398600.4418 \text{ km}^3 s^{-2}$ is the standard gravitational parameter of the Earth, $J_2 = 1082.63 \times 10^{-6}$ defines the oblateness perturbation, $r_{eq} = 6378.1370 \text{ km}$ is the Earth equatorial radius.

Invariant Solution Strategy

A two-part solution strategy is presented. First, two Lagrange-Like invariants are introduced to automate the process of generating arbitrary order time derivatives for the equation of motion in Eq. (2). The first scalar Lagrange-like invariant is:

$$f = \mathbf{r}.\mathbf{r} \tag{4}$$

where (.) denotes the vector inner product for the bilinear form. The n-th order time derivative of f is computed by applying *Leibniz product rule* as,

$$f^{(n)} = \sum_{m=0}^{n} \binom{n}{m} \mathbf{r}^{(m)}.\mathbf{r}^{(n-m)} \tag{5}$$

where $\binom{n}{m}$ is the standard binomial and $\frac{d^m f}{dt^m} = f^{(m)}$. The second Lagrange-Like invariant is given by,

$$g_p = f^{-p/2} \tag{6}$$

which is not a bilinear form, however, the first time derivative for Eq. (6) can be shown to be the bilinear form: $fg_p + \dfrac{p}{2} f g_p = 0$. Evaluating the higher order time derivatives of g_p, using *Leibniz product rule*, and solving for the highest time derivative, one obtains,

$$g_p^{(n+1)} = -\left\{ \frac{p}{2} f^{(1)} g^{(n)} + \sum_{m=1}^{n} \binom{n}{m} \left(\frac{p}{2} f^{(m+1)} g^{(n-m)} + f^{(m)} g^{(n-m+1)} \right) \right\} / f \qquad (7)$$

Note, that if the bilinear first derivative term was not used for Leibnitz product rule, that one is forced to model Eq. (6) as a composite function of the form $g_p = g_p(f)$, which is classically handled by the very complicated and celebrated *Faá di Bruno* mathematical identity.

From Eq. (3), it is obvious that $\mathbf{a}_{J_2}$ contains cubic and quartic terms that are not handled by Leibnitz product rule. Theoretically, it is possible to develop multi-nominal versions of Leibnitz product rule; nevertheless, by introducing several variable transformations, $\mathbf{a}_{J_2}$ is recast in a form suitable for applying the binomial form of Leibnitz product rule for developing arbitrary-order analytic time derivative models. To this end, the first step is to define the constant

$$c_2 = \frac{3}{2} J_2 \mu r_{eq}^2 \qquad (8)$$

Now Eq. (2) is re-written as

$$\mathbf{r} = -\mu \mathbf{r} (\mathbf{r} \cdot \mathbf{r})^{-3/2} - c_2 \left(\frac{x}{r^5} - 5\frac{z^2 x}{r^7}, \quad \frac{y}{r^5} - 5\frac{z^2 y}{r^7}, \quad 3\frac{z}{r^5} - 5\frac{z^3}{r^7} \right) \qquad (9)$$

Recalling Eq. (6), Eq. (9) becomes

$$\mathbf{r} = -\mu \mathbf{r} g_3 - c_2 \begin{pmatrix} xg_5 - 5z^2 xg_7 \\ yg_5 - 5z^2 yg_7 \\ 3zg_5 - 5z^3 g_7 \end{pmatrix} \qquad (10)$$

New recursions are developed in Appendix C for the quartic terms appearing in Eq. (10) to enable the use of Leibnitz Product rule. With these intermediate variable definitions the $\mathbf{a}_{J_2}$ perturbed acceleration is expressed in the desired bi-linear product form:

$$\mathbf{r} = -\mu \mathbf{r} g_3 - c_2 \begin{pmatrix} xg_5 - 5\phi g_7 \\ yg_5 - 5\varphi g_7 \\ 3zg_5 - 5\psi g_7 \end{pmatrix} \qquad (11)$$

The nth-order time derivative is obtained by using *Leibnitz product rule*, as follows:

$$r^{(n+2)} = -\mu \sum_{m=0}^{n} \binom{n}{m} r^{(m)} g_3^{(n-m)} - c_2 \sum_{m=0}^{n} \binom{n}{m} \begin{pmatrix} x^{(m)} g_5^{(n-m)} - 5\phi^{(m)} g_7^{(n-m)} \\ y^{(m)} g_5^{(n-m)} - 5\varphi^{(m)} g_7^{(n-m)} \\ 3z^{(m)} g_5^{(n-m)} - 5\psi^{(m)} g_7^{(n-m)} \end{pmatrix} \tag{12}$$

The complete power series time derivative models are computed by recursively combining Eqs. (5), (7), (11)-(12), and evaluating the series for the position and velocity vectors,

$$r(t+h) = r(t) + r(t)h + r(t)h^2/2! + r(t)h^3/3! +$$

$$r(t+h) = r(t) + r(t)h + r(t)h^2/2! +$$

The two-body solution is recovered by setting $c_2 = 0$.

STATE TRANSITION TENSORS

The development of state transition tensor models demands that previously derived recursive equations must be generalized to account for initial condition partial derivative calculations. The generalized partial derivative models for *f* and *g* are presented in Appendices *A* and *B*, respectively. In every case the application of *Leibnitz product rule* provides a systematic procedure for building all of the required partials. In this way one is left with dealing with evaluating partials for Eqs.(11) and (12). The first step in the process consists of casting the system dynamics in state-space form:

$$x_1 = r \quad x_2 = r \quad \Rightarrow \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} x_2 \\ -\mu x_1 g_3 + a_{J2} \end{pmatrix} \tag{13}$$

Two steps are required for developing the state transition tensor models. First, one evaluates the nth order time derivative of Eq. (13), leading to

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix}^{(n)} = \begin{pmatrix} x_2^{(n)} \\ \sum_{m=0}^{n} \binom{n}{m} \left(-\mu x_1^{(m)} g_3^{(n-m)} \right) + a_{J2}^{(n)} \end{pmatrix} \tag{14}$$

Second, one sequentially evaluates the state transition tensor spatial partial derivatives. The first- through third-order STTS for Eq. (13) follow as

First-Order State Transition Matrix

N-th Order time derivatives of this model are developed by applying *Leibnitz product rule*, as follows:

$$\binom{x_1}{x_2}_{,i}^{(n)} = \begin{pmatrix} x_{2,i}^{(n)} \\ \sum_{m=0}^{n}\binom{n}{m}\left(-\mu x_{1,i}^{(m)}g^{(n-m)} - \mu x_1^{(m)}g_{,i}^{(n-m)}\right) + a_{J2,i}^{(n)} \end{pmatrix}$$

$$a_{J2,i}^{(n)} = -c_2 \sum_{m=0}^{n}\binom{n}{m}\begin{pmatrix} x_{,i}^{(m)}g_5^{(n-m)} + x^{(m)}g_{5,i}^{(n-m)} - 5\phi_{,i}^{(m)}g_7^{(n-m)} - 5\phi^{(m)}g_{7,i}^{(n-m)} \\ y_{,i}^{(m)}g_5^{(n-m)} + y^{(m)}g_{5,i}^{(n-m)} - 5\varphi_{,i}^{(m)}g_7^{(n-m)} - 5\varphi^{(m)}g_{7,i}^{(n-m)} \\ 3z_{,i}^{(m)}g_5^{(n-m)} + 3z^{(m)}g_{5,i}^{(n-m)} - 5\psi_{,i}^{(m)}g_7^{(n-m)} - 5\psi^{(m)}g_{7,i}^{(n-m)} \end{pmatrix}$$

(15)

Second-Order State Transition Tensor

N-th Order time derivatives of this model are developed by applying *Leibnitz product rule*, as follows:

$$\binom{x_1}{x_2}_{,ij}^{(n)} = \begin{pmatrix} x_{2,ij}^{(n)} \\ \sum_{m=0}^{n}\binom{n}{m}\left(-\mu x_{1,ij}^{(m)}g^{(n-m)} - \mu x_{1,i}^{(m)}g_{,j}^{(n-m)} - \mu x_{1,j}^{(m)}g_{,i}^{(n-m)} - \mu x_1^{(m)}g_{,ij}^{(n-m)}\right) + a_{J2,ij}^{(n)} \end{pmatrix}$$

(16)

$$a_{J2,ij}^{(n)} = -c_2 \sum_{m=0}^{n}\binom{n}{m}\begin{pmatrix} x_{,ij}^{(m)}g_5^{(n-m)} + x_{,i}^{(m)}g_{5,j}^{(n-m)} + x_{,j}^{(m)}g_{5,i}^{(n-m)} + x^{(m)}g_{5,ij}^{(n-m)} - \\ 5\phi_{,ij}^{(m)}g_7^{(n-m)} - 5\phi_{,i}^{(m)}g_{7,j}^{(n-m)} - 5\phi_{,j}^{(m)}g_{7,i}^{(n-m)} - 5\phi^{(m)}g_{7,ij}^{(n-m)} \\ \\ y_{,ij}^{(m)}g_5^{(n-m)} + y_{,i}^{(m)}g_{5,j}^{(n-m)} + y_{,j}^{(m)}g_{5,i}^{(n-m)} + y^{(m)}g_{5,ij}^{(n-m)} - \\ 5\varphi_{,ij}^{(m)}g_7^{(n-m)} - 5\varphi_{,i}^{(m)}g_{7,j}^{(n-m)} - 5\varphi_{,j}^{(m)}g_{7,i}^{(n-m)} - 5\varphi^{(m)}g_{7,ij}^{(n-m)} \\ \\ 3z_{,ij}^{(m)}g_5^{(n-m)} + 3z_{,i}^{(m)}g_{5,j}^{(n-m)} + 3z_{,j}^{(m)}g_{5,i}^{(n-m)} + 3z^{(m)}g_{5,ij}^{(n-m)} - \\ 5\psi_{,ij}^{(m)}g_7^{(n-m)} - 5\psi_{,i}^{(m)}g_{7,j}^{(n-m)} - 5\psi_{,j}^{(m)}g_{7,i}^{(n-m)} - 5\psi^{(m)}g_{7,ij}^{(n-m)} \end{pmatrix}$$

Third-Order State Transition Tensor

N-th Order time derivatives of this model are developed by applying *Leibnitz product rule*, as follows:

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix}_{,ijk} = \begin{pmatrix} x_2 \\ -\mu x_1 g + a_{J2} \end{pmatrix}_{,ijk}^{(n)}$$

$$= \begin{pmatrix} x_{2,ijk}^{(n)} \\ (m) \\ -\mu \sum_{m=0}^{n} \binom{n}{m} \begin{pmatrix} x_{1,ijk}^{(m)} g^{(n-m)} + x_{1,ij}^{(m)} g_{,k}^{(n-m)} + x_{1,ik}^{(m)} g_{,j}^{(n-m)} + x_{1,i}^{(m)} g_{,jk}^{(n-m)} + \\ x_{1,jk}^{(m)} g_{,i}^{(n-m)} + x_{1,j}^{(m)} g_{,ik}^{(n-m)} + x_{1,k}^{(m)} g_{,ij}^{(n-m)} + x_1^{(m)} g_{,ijk}^{(n-m)} \end{pmatrix} + a_{J2,ijk}^{(n)} \end{pmatrix}$$

$$a_{J2,ijk}^{(n)} = -c_2 \sum_{m=0}^{n} \binom{n}{m} \begin{pmatrix} x_{,ijk}^{(m)} g_5^{(n-m)} + x_{,ij}^{(m)} g_{5,k}^{(n-m)} + x_{,ik}^{(m)} g_{5,j}^{(n-m)} + x_{,i}^{(m)} g_{5,jk}^{(n-m)} + \\ x_{,jk}^{(m)} g_{5,i}^{(n-m)} + x_{,j}^{(m)} g_{5,ik}^{(n-m)} + x_{,k}^{(m)} g_{5,ij}^{(n-m)} + x^{(m)} g_{5,ijk}^{(n-m)} - \\ 5\phi_{,ijk}^{(m)} g_7^{(n-m)} - 5\phi_{,ij}^{(m)} g_{7,k}^{(n-m)} - 5\phi_{,ik}^{(m)} g_{7,j}^{(n-m)} - 5\phi_{,i}^{(m)} g_{7,jk}^{(n-m)} - \\ 5\phi_{,jk}^{(m)} g_{7,i}^{(n-m)} - 5\phi_{,j}^{(m)} g_{7,ik}^{(n-m)} - 5\phi_{,k}^{(m)} g_{7,ij}^{(n-m)} - 5\phi^{(m)} g_{7,ijk}^{(n-m)} \\[2mm] y_{,ijk}^{(m)} g_5^{(n-m)} + y_{,ij}^{(m)} g_{5,k}^{(n-m)} + y_{,ik}^{(m)} g_{5,j}^{(n-m)} + y_{,i}^{(m)} g_{5,jk}^{(n-m)} + \\ y_{,jk}^{(m)} g_{5,i}^{(n-m)} + y_{,j}^{(m)} g_{5,ik}^{(n-m)} + y_{,k}^{(m)} g_{5,ij}^{(n-m)} + y^{(m)} g_{5,ijk}^{(n-m)} - \\ 5\varphi_{,ijk}^{(m)} g_7^{(n-m)} - 5\varphi_{,ij}^{(m)} g_{7,k}^{(n-m)} - 5\varphi_{,ik}^{(m)} g_{7,j}^{(n-m)} - 5\varphi_{,i}^{(m)} g_{7,jk}^{(n-m)} - \\ 5\varphi_{,jk}^{(m)} g_{7,i}^{(n-m)} - 5\varphi_{,j}^{(m)} g_{7,ik}^{(n-m)} - 5\varphi_{,k}^{(m)} g_{7,ij}^{(n-m)} - 5\varphi^{(m)} g_{7,ijk}^{(n-m)} \\[2mm] 3z_{,ijk}^{(m)} g_5^{(n-m)} + 3z_{,ij}^{(m)} g_{5,k}^{(n-m)} + 3z_{,ik}^{(m)} g_{5,j}^{(n-m)} + 3z_{,i}^{(m)} g_{5,jk}^{(n-m)} + \\ 3z_{,jk}^{(m)} g_{5,i}^{(n-m)} + 3z_{,j}^{(m)} g_{5,ik}^{(n-m)} + 3z_{,k}^{(m)} g_{5,ij}^{(n-m)} + 3z^{(m)} g_{5,ijk}^{(n-m)} - \\ 5\psi_{,ijk}^{(m)} g_7^{(n-m)} - 5\psi_{,ij}^{(m)} g_{7,k}^{(n-m)} - 5\psi_{,ik}^{(m)} g_{7,j}^{(n-m)} - 5\psi_{,i}^{(m)} g_{7,jk}^{(n-m)} - \\ 5\psi_{,jk}^{(m)} g_{7,i}^{(n-m)} - 5\psi_{,j}^{(m)} g_{7,ik}^{(n-m)} - 5\psi_{,k}^{(m)} g_{7,ij}^{(n-m)} - 5\psi^{(m)} g_{(n-m)} \end{pmatrix} \tag{17}$$

A key observation is that all terms above remain bilinear, this means that arbitrary order time derivative models are easily derived. As a result, all of the first- through third-order state transition tensors can be recursive power series solutions coupled to the recursive solution for the state trajectories. The two-body problem state transition tensor are obtained by setting $c_2 = 0$. From the bilinear data structure above, it is obvious that all the terms in the state and state transition tensors can be computed in a massively parallel way.

As with any power series-based approximation one must come to grips with the issues: (1) how many terms need to be retained in the approximations, (2) how large can the step-size h be made for maintaining what level of position and velocity accuracy, (3) can variable step-size algorithms be developed for accelerating the series approximations, and (4) what is the computational performance of the series approximation when compared to other available approximation methods. The recursive nature of problem formulation is well suited for parallel implementations.

1177

NUMERICAL CASE STUDIES

Two types of orbits, 2D and 3D, are presented here as numerical case studies for the analytic continuation method. Each case is integrated for one orbital period for the unperturbed and the J_2 perturbed orbits. The analysis compares each case with existing numerical techniques, MATLAB *ODE45* and *RK1210*, in terms of speed and accuracy. For the unperturbed orbit the classical F&G solution is used as the truth for accuracy calculations whereas for the perturbed orbit the methods are compared relative to each other.

Case1: 2D Orbit

The initial conditions for the orbit are given by

$$\mathbf{r}_0 = \begin{bmatrix} 7 & 0 & 0 \end{bmatrix}^T \times 10^6 \text{ m}$$

$$\mathbf{v}_0 = \begin{bmatrix} 0 & 7.7324 & 0 \end{bmatrix}^T \times 10^3 \text{ m/s}$$

For the unperturbed orbit, at each time step the error between the analytic continuation solution, subscript AC, and the classical F&G, subscript FG, is computed as,

$$\mathbf{r}_{FG} - \mathbf{r}_{AC} = \begin{bmatrix} \Delta x_0 & \Delta y_0 & \Delta z_0 \end{bmatrix}^T$$

$$\mathbf{v}_{FG} - \mathbf{v}_{AC} = \begin{bmatrix} \Delta x_0 & \Delta y_0 & \Delta z_0 \end{bmatrix}^T$$

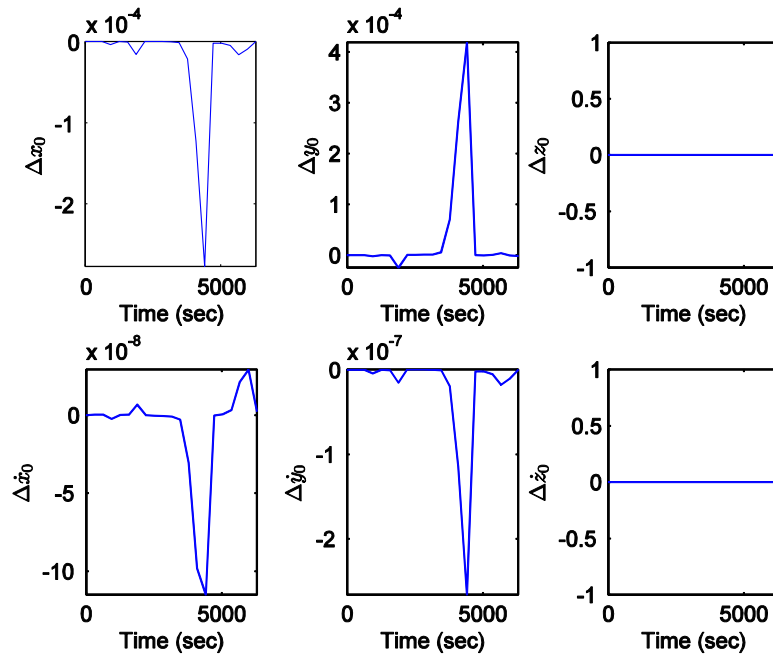Figure 1 shows the errors for all 6 states for one orbital period.

Figure 1: Analytic Continuation vs. F&G Solution Case1

The same errors calculation is repeated for the two numerical integrators, *ODE45* and *RK1210*, in order to ensure that the same level of accuracy is obtained for all methods, Figure 2 and Figure 3.
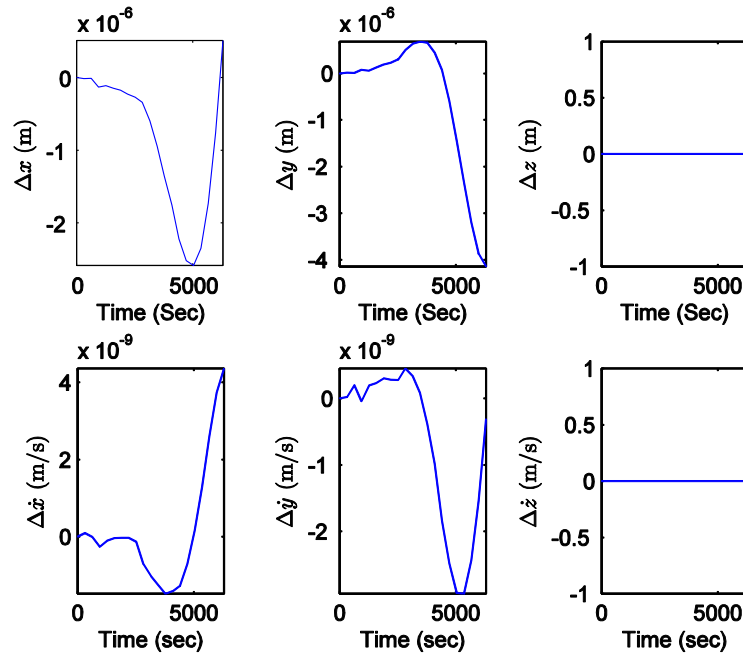
1178

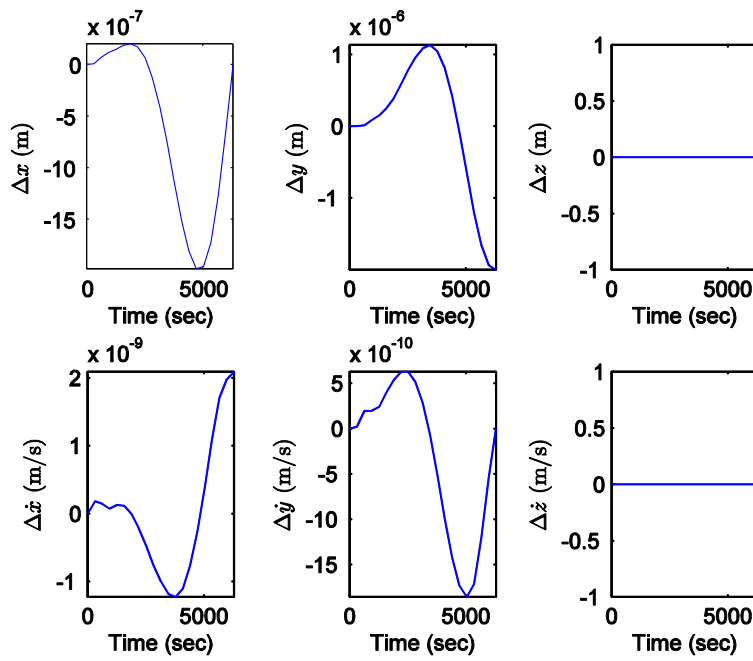**Figure 2:** *ODE45* **vs. F&G Solution Case 1**

Figure 3: *RK1210* **vs. F&G Solution Case 1**

Introducing the J_2 term to the orbit *Analytic Continuation*, *ODE45* and *RK1210* are used to compute the trajectory. As the F&G classical solution is not feasible to use due to the orbit perturbation the three methods are compared against each other for accuracies and timings. The accuracies criterion is achieving sub meter relative errors in position between the three methods. First the *Analytic Continuation* is compared against *ODE45*, Figure 4. The method is then compared against *RK1210*, Figure 5. Finally the simulation timing comparison is shown in Figure 6.
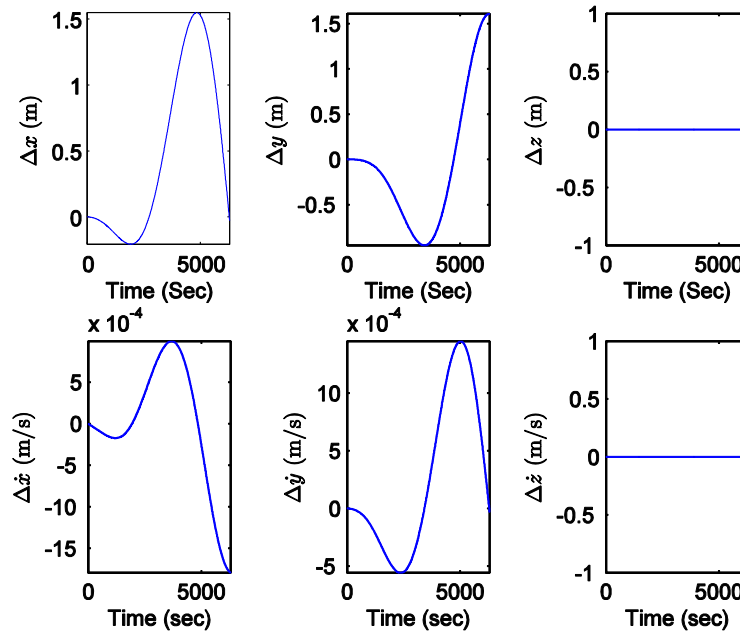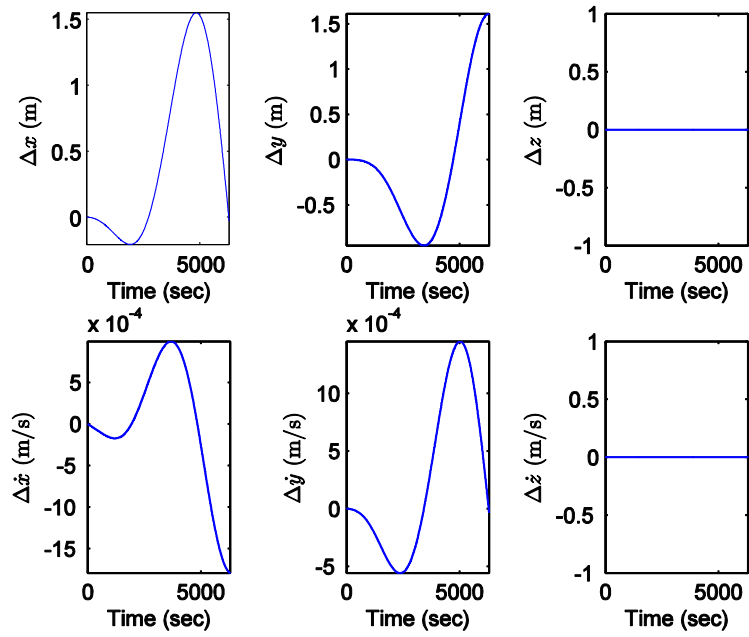

Figure 4: Analytic Continuation vs. *ODE45* Case1 J2

1180

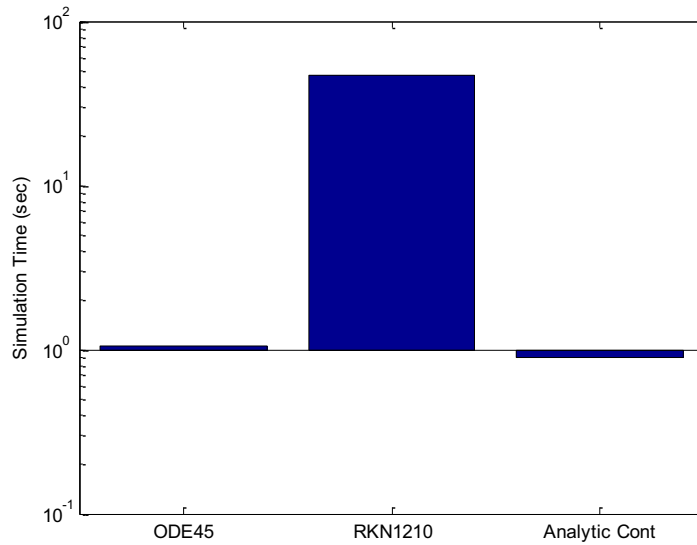**Figure 5: Analytic Continuation vs. *RK1210* Case1 J2**

Figure 6: Simulation Time Comparison Case1 J2

For sub meter relative accuracies *Analytic Continuation* approach is again far more superior to *RK1210*, at least one order of magnitude, whereas it is slightly faster than *ODE45*, approximately 2X. For a fixed step size algorithm compared against highly optimized variable step size methods the speed up is quite significant and can be extended further by either introducing a variable step size technique or using parallel computing.

Case2: 3D Orbit

The initial conditions for the orbit are given by

$$\mathbf{r}_0 = \begin{bmatrix} 1.058 & -6.148 & 2.086 \end{bmatrix}^T \times 10^6 \text{ m}$$

$$\mathbf{v}_0 = \begin{bmatrix} 2.658 & 2.77 & -6.815 \end{bmatrix}^T \times 10^3 \text{ m/s}$$

As in case1 for the unperturbed orbit the comparison against the classical F&G solution is shown in Figure 9.
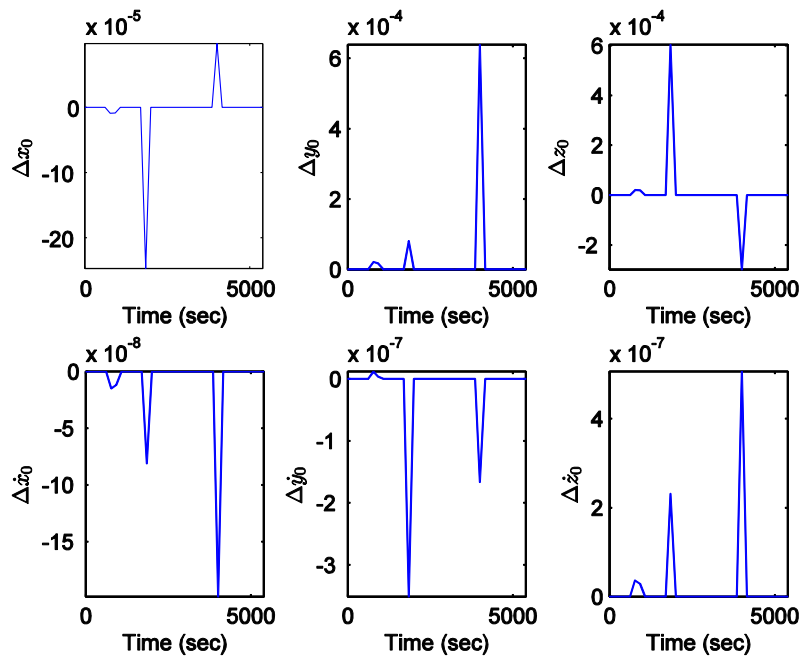

Figure 7: Analytic Continuation vs. F&G Solution Case2

The same errors calculation is repeated for the two numerical integrators, *ODE45* and *RK1210*, used in order to ensure that the same level of accuracy is obtained for all method, Figure 8 and Figure 9.
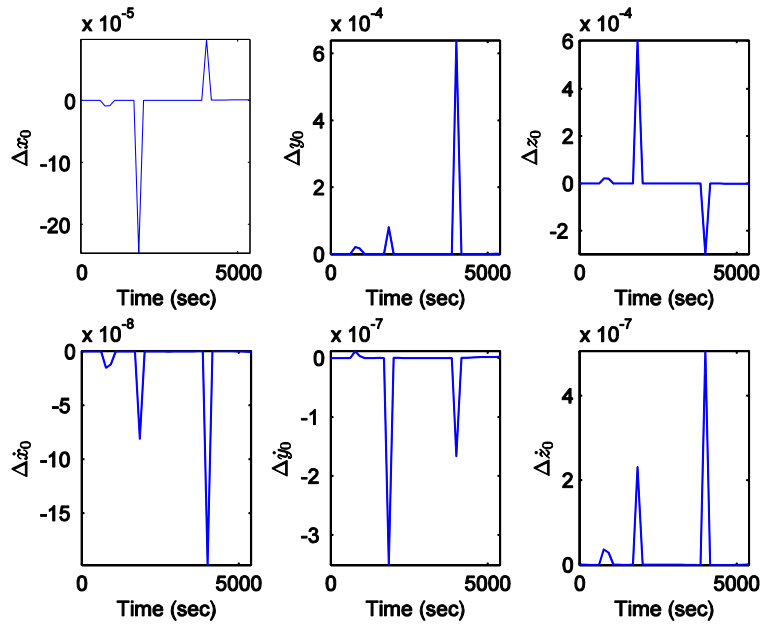
Figure 8: *ODE45* **vs. F&G Solution Case2**
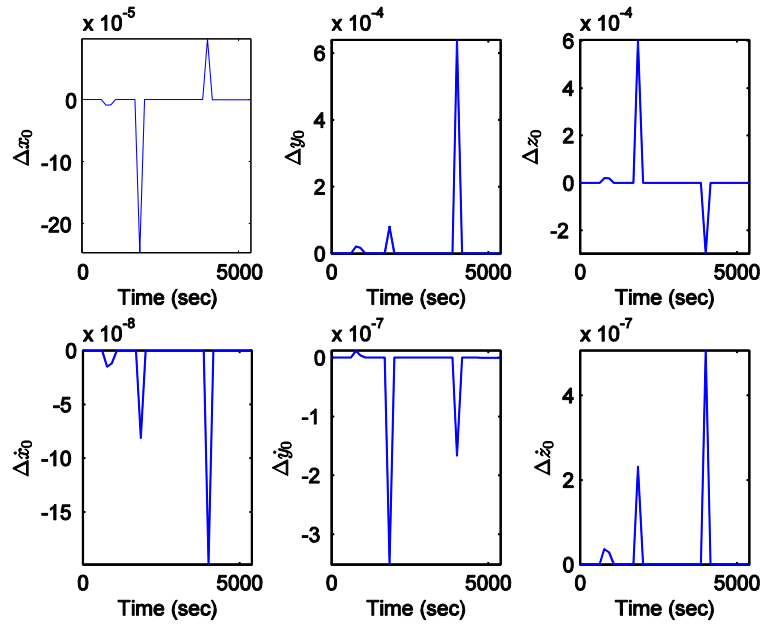

Figure 9: *RK1210* **vs. F&G Solution Case2**

As the same level of accuracy is achieved by all three methods the simulation time for each method is calculated and plotted, Figure 10. It is quite obvious that for the same level of accuracy the analytic continuation is exceedingly superior to the existing numerical methods with at least an order of magnitude speed up in simulation time.
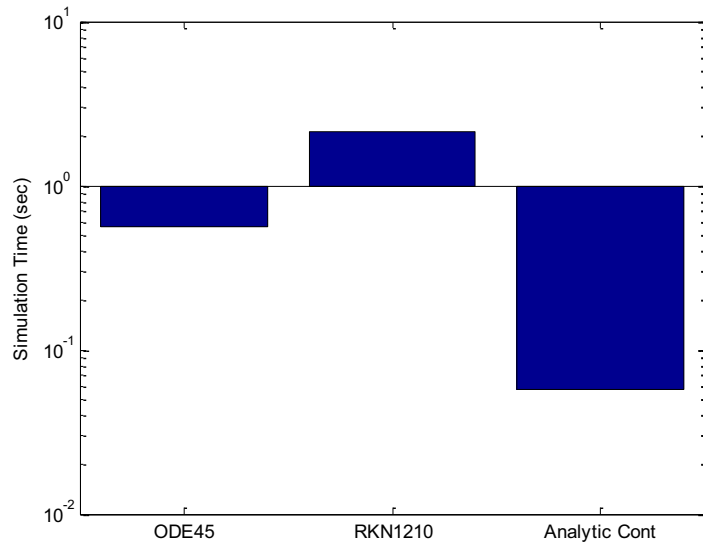
Figure 10: Simulation Time Comparison Case2 Unperturbed Orbit

The J2 effect is now introduced to the orbit. In this case also the classical F&G solution is not used as the truth and the three methods will be compared against each other in terms of accuracies and timings. *Analytic Continuation* vs. *ODE45* and *Analytic Continuation* vs. *RK1210* accuracies comparisons are shown in Figure 11 and Figure 12 respectively for a half orbital period to reasonably reduce the total simulation time for the three methods combined. The timings comparison for the three methods is shown in Figure 13.
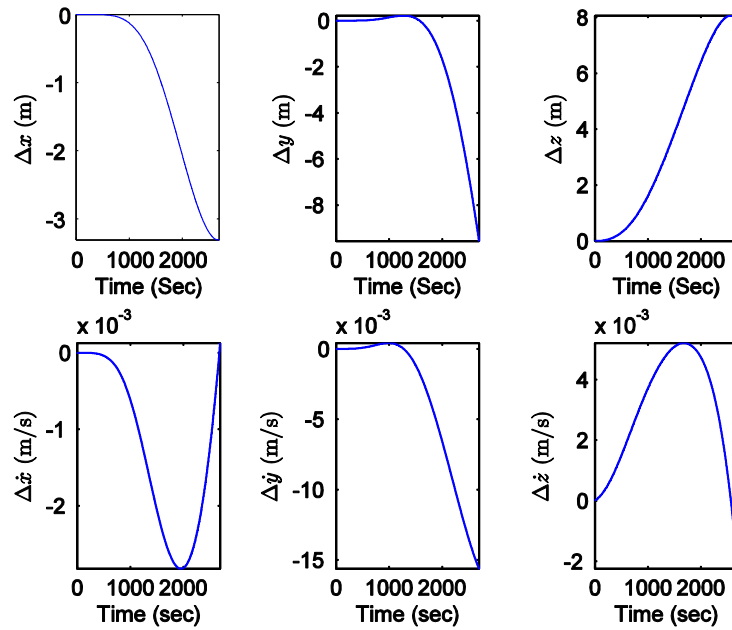
Figure 11: *Analytic continuation* vs. *ODE45* Case2 J2

1184
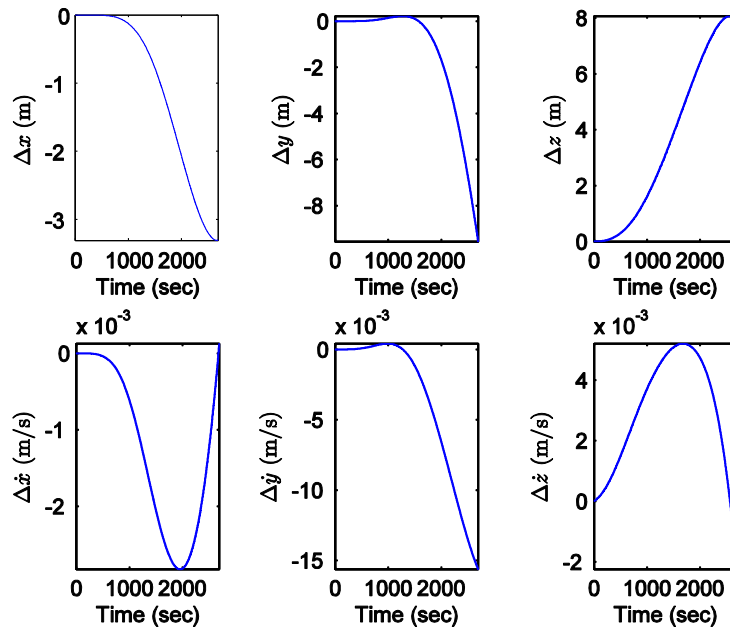
Figure 12: *Analytic Continuation* **vs.** *RK1210* **Case2 J2**
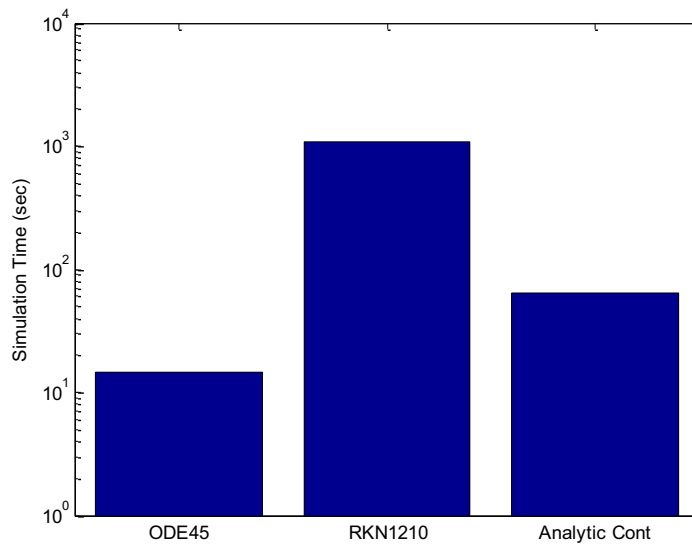
Figure 13: Simulation Time Comparison Case2 J2

The *Analytic Continuation* method is about an order of magnitude faster than *RK1210* whereas when compared against *ODE45* it came out 4X slower. The advantage of the variable step size of *ODE45* is quite significant in this example and a significant improvement to the *Analytic Continuation* speed can be achieved by implementing a similar variable step algorithm or using parallel computing.

1185

DISCUSSION AND CONCLUSION

The analytic continuation solution is extended to solve a broader case of problems to encompass the perturbed two-body problem for the Earth oblateness effects. This solution scheme can be easily extended to evaluate the higher perturbation harmonics as function of the predefined Lagrange-Like invariants. The speed and accuracy enabled by the power series solution is orders of magnitude better than the current standard numerical integrators in terms of speed while retaining the same accuracies.

The Analytic continuation solution technique is also used to derive the state transition tensors for the perturbed two body problem up to third order. Numerical results for perturbed and unperturbed 2D and 3D orbits are presented. Accuracies and timings comparisons are shown against existing numerical methods, *RK1210* and *ODE45*. For the same level of accuracy the *Analytic Continuation* method showed significant superiority in terms of speed over *RK1210* whereas it produced very similar timings to the more optimized *ODE45*. As a future development variable step size can be introduced to the method and parallel computing can be easily implemented for more speed gains.

REFERENCES

[1]   *Junkins, J. L., Majji, M., and Turner, J. D.* "*High* Order Keplerian State Transition Tensors," *In Proceedings of the F. Landis Markley Astronautics Symposium, 2008.* AAS 08-270.

[2]   Lagrange, J. L. Mécanique Analytique. Paris: Gauthier-Villars, 1811.

[3]   Battin, R. H. An introduction to the mathematics and methods of astrodynamics, AIAA, 1999.

[4]   Turner, J. D., Elgohary, T., Majji, M., and Junkins, J. "High Accuracy Trajectory and Uncertainty Propogation Algorithm for Long-Term Asteroid Motion Prediction," Adventures on the interface of Mechanics and Control. Tech Science Press, 2012.

[5]   Elgohary, T., Turner, J. D., and Junkins, J. L. "High-Order Analytic Continuation and Numerical Stability Analysis for the Classical Two-Body Problem," The Jer-Nan Juan Astrodynamics Symposium, 2012. AAS 12-639.

[6]   Brouwer, D., and Clemence, G. M. Methods of celestial mechanics, New York Academic Press, 1961 Vol. 1, 1961.

[7]   Hamilton, W. R. "On a general method in dynamics," Philosophical Transactions of the Royal Society Vol. 2, 1834, pp. 247-308.

[8]   Schaub, H., and Junkins, J. L. Analytical mechanics of space systems, AIAA, 2003.

APPENDIX A: CHANGE OF VARIABLE f FUNCTION TIME AND PARTIAL DERIVATIVES

The scalar f function is defined as

$$f = \boldsymbol{r} \cdot \boldsymbol{r} \tag{18}$$

and the first- through fourth-order partial derivatives follow as

$$f_{,i} = 2\boldsymbol{r} \cdot \boldsymbol{r}_{,i}$$

$$f_{,ij} = 2\boldsymbol{r}_{,j} \cdot \boldsymbol{r}_{,i} + 2\boldsymbol{r} \cdot \boldsymbol{r}_{,ij}$$

$$f_{,ijk} = 2\boldsymbol{r}_{,jk} \cdot \boldsymbol{r}_{,i} + 2\boldsymbol{r}_{,j} \cdot \boldsymbol{r}_{,ik} + 2\boldsymbol{r}_{,k} \cdot \boldsymbol{r}_{,ij} + 2\boldsymbol{r} \cdot \boldsymbol{r}_{,ijk} \tag{19}$$

$$f_{,ijkl} = 2\boldsymbol{r}_{,jkl} \cdot \boldsymbol{r}_{,i} + 2\boldsymbol{r}_{,jk} \cdot \boldsymbol{r}_{,il} + 2\boldsymbol{r}_{,jl} \cdot \boldsymbol{r}_{,ik} + 2\boldsymbol{r}_{,j} \cdot \boldsymbol{r}_{,ikl} +$$
$$2\boldsymbol{r}_{,kl} \cdot \boldsymbol{r}_{,ij} + 2\boldsymbol{r}_{,k} \cdot \boldsymbol{r}_{,ijl} + 2\boldsymbol{r}_{,l} \cdot \boldsymbol{r}_{,ijk} + 2\boldsymbol{r} \cdot \boldsymbol{r}_{,ijkl}$$

The first- through fourth-order time derivatives follow as

$$f^{(n)} = \sum_{m=0}^{n} \binom{n}{m} \boldsymbol{r}^{(m)} \cdot \boldsymbol{r}^{(n-m)}$$

$$f_{,i}^{(n)} = 2\sum_{m=0}^{n} \binom{n}{m} \boldsymbol{r}^{(m)} \cdot \boldsymbol{r}_{,i}^{(n-m)}$$

$$f_{,ij}^{(n)} = 2\sum_{m=0}^{n} \binom{n}{m} \boldsymbol{r}_{,j}^{(m)} \cdot \boldsymbol{r}_{,i}^{(n-m)} + 2\sum_{m=0}^{n} \binom{n}{m} \boldsymbol{r}^{(m)} \cdot \boldsymbol{r}_{,ij}^{(n-m)}$$

$$f_{,ijk}^{(n)} = 2\sum_{m=0}^{n} \binom{n}{m} \boldsymbol{r}_{,jk}^{(m)} \cdot \boldsymbol{r}_{,i}^{(n-m)} + 2\sum_{m=0}^{n} \binom{n}{m} \boldsymbol{r}_{,j}^{(m)} \cdot \boldsymbol{r}_{,ik}^{(n-m)} +$$
$$2\sum_{m=0}^{n} \binom{n}{m} \boldsymbol{r}_{,k}^{(m)} \cdot \boldsymbol{r}_{,ij}^{(n-m)} + 2\sum_{m=0}^{n} \binom{n}{m} \boldsymbol{r}^{(m)} \cdot \boldsymbol{r}_{,ijk}^{(n-m)}$$

$$f_{,ijkl}^{(n)} = 2\sum_{m=0}^{n} \binom{n}{m} \boldsymbol{r}_{,jkl}^{(m)} \cdot \boldsymbol{r}_{,i}^{(n-m)} + 2\sum_{m=0}^{n} \binom{n}{m} \boldsymbol{r}_{,jk}^{(m)} \cdot \boldsymbol{r}_{,il}^{(n-m)} +$$
$$2\sum_{m=0}^{n} \binom{n}{m} \boldsymbol{r}_{,jl}^{(m)} \cdot \boldsymbol{r}_{,ik}^{(n-m)} + 2\sum_{m=0}^{n} \binom{n}{m} \boldsymbol{r}_{,j}^{(m)} \cdot \boldsymbol{r}_{,ikl}^{(n-m)} +$$
$$2\sum_{m=0}^{n} \binom{n}{m} \boldsymbol{r}_{,kl}^{(m)} \cdot \boldsymbol{r}_{,ij}^{(n-m)} + 2\sum_{m=0}^{n} \binom{n}{m} \boldsymbol{r}_{,k} \cdot \boldsymbol{r}_{,ijl} + \tag{20}$$
$$2\sum_{m=0}^{n} \binom{n}{m} \boldsymbol{r}_{,l}^{(m)} \cdot \boldsymbol{r}_{,ijk}^{(n-m)} + 2\sum_{m=0}^{n} \binom{n}{m} \boldsymbol{r}^{(m)} \cdot \boldsymbol{r}_{,ijkl}^{(n-m)}$$

APPENDIX B: CHANGE OF VARIABLE g FUNCTION TIME AND PARTIAL DERIVATIVES

The g function is defined as

$$g = f^{-3/2} \tag{1}$$

In anticipation of supporting the J2 calculations, one recognizes that Eq. (1) is generalized as follows:

$$g(\sigma) = f^{-\sigma/2} \tag{2}$$

where σ can assume the values $(3, \quad 5, \quad 7)$. What is of critical importance for calculations is that both differential and partial differential equations for Eq. (2) are cast in the form of bi-linear products, without denominators, which leads to very simple sensitivity equations.

Partial Derivative Calculations

The first four partial derivatives are sequentially obtained follow as

$$0 = \sigma g f_{,i} + 2 f g_{,i}$$

$$\Rightarrow g_{,i} = -\frac{\sigma g f_{,i}}{2f}$$

$$0 = \sigma g_{,j} f_{,i} + \sigma g f_{,ij} + 2 f_{,j} g_{,i} + 2 f g_{,ij}$$

$$\Rightarrow g_{,ij} = -\left(\sigma g_{,j} f_{,i} + \sigma g f_{,ij} + 2 f_{,j} g_{,i}\right)/\left(2f\right)$$

$$0 = \sigma g_{,jk} f_{,i} + \sigma g_{,j} f_{,ik} + \sigma g_{,k} f_{,ij} + \sigma g f_{,ijk} +$$
$$2 f_{,jk} g_{,i} + 2 f_{,j} g_{,ik} + 2 f_{,k} g_{,ij} + 2 f g_{,ijk}$$

$$\Rightarrow g_{,ijk} = -\left(\begin{matrix} \sigma g_{,jk} f_{,i} + \sigma g_{,j} f_{,ik} + \sigma g_{,k} f_{,ij} + \sigma g f_{,ijk} + \\ 2 f_{,jk} g_{,i} + 2 f_{,j} g_{,ik} + 2 f_{,k} g_{,ij} \end{matrix} \right)/\left(2f\right)$$

$$0 = \sigma g_{,jkl} f_{,i} + \sigma g_{,jk} f_{,il} + \sigma g_{,jl} f_{,ik} + \sigma g_{,j} f_{,ikl} +$$
$$\sigma g_{,kl} f_{,ij} + \sigma g_{,k} f_{,ijl} + \sigma g_{,l} f_{,ijk} + \sigma g f_{,ijkl} +$$
$$2 f_{,jkl} g_{,i} + 2 f_{,jk} g_{,il} + 2 f_{,jl} g_{,ik} + 2 f_{,j} g_{,ikl} +$$
$$2 f_{,kl} g_{,ij} + 2 f_{,k} g_{,ijl} + 2 f_{,l} g_{,ijk} + 2 f g_{,ijkl}$$

$$\Rightarrow g_{,ijkl} = -\left(\begin{matrix} \sigma g_{,jkl} f_{,i} + \sigma g_{,jk} f_{,il} + \sigma g_{,jl} f_{,ik} + \sigma g_{,j} f_{,ikl} + \\ \sigma g_{,kl} f_{,ij} + \sigma g_{,k} f_{,ijl} + \sigma g_{,l} f_{,ijk} + \sigma g f_{,ijkl} + \\ 2 f_{,jkl} g_{,i} + 2 f_{,jk} g_{,il} + 2 f_{,jl} g_{,ik} + 2 f_{,j} g_{,ikl} + \\ 2 f_{,kl} g_{,ij} + 2 f_{,k} g_{,ijl} + 2 f_{,l} g_{,ijk} \end{matrix} \right)/\left(2f\right)$$

(3)

Time Derivative Calculations

In all of the time derivative calculations, one needs to observe that the final summation starts at one rather than zero, this results from solving for the highest time derivative in each case. The nth order time derivatives for the first order partial derivative follow as

$$0 = \sigma \sum_{m=0}^{n} \binom{n}{m} g^{(m)} f_{,i}^{(n-m)} + 2 \sum_{m=0}^{n} \binom{n}{m} f^{(m)} g_{,i}^{(n-m)}$$

(4)

$$\Rightarrow g_{,i}^{(n)} = -\left\{ \sigma \sum_{m=0}^{n} \binom{n}{m} g^{(m)} f_{,i}^{(n-m)} + 2 \sum_{m=1}^{n} \binom{n}{m} f^{(m)} g_{,i}^{(n-m)} \right\}/2 f^{(0)}$$

The nth order time derivatives for the second order partial derivative follow as

$$0 = \sigma \sum_{m=0}^{n} \binom{n}{m} g_{,j}^{(m)} f_{,i}^{(n-m)} + \sigma \sum_{m=0}^{n} \binom{n}{m} g^{(m)} f_{,ij}^{(n-m)} +$$

$$2 \sum_{m=0}^{n} \binom{n}{m} f_{,j}^{(m)} g_{,i}^{(n-m)} + 2 \sum_{m=0}^{n} \binom{n}{m} f^{(m)} g_{,ij}^{(n-m)}$$

$$(5)$$

$$\Rightarrow g_{,ij}^{(n)} = - \left\{ \begin{array}{l} \sigma \sum_{m=0}^{n} \binom{n}{m} g_{,j}^{(m)} f_{,i}^{(n-m)} + \sigma \sum_{m=0}^{n} \binom{n}{m} g^{(m)} f_{,ij}^{(n-m)} + \\ 2 \sum_{m=0}^{n} \binom{n}{m} f_{,j}^{(m)} g_{,i}^{(n-m)} + 2 \sum_{m=1}^{n} \binom{n}{m} f^{(m)} g_{,ij}^{(n-m)} \end{array} \right\} / 2 f^{(0)}$$

The nth order time derivatives of the third order partial derivative follow as

$$0 = \sigma \sum_{m=0}^{n} \binom{n}{m} g_{,jk}^{(m)} f_{,i}^{(n-m)} + \sigma \sum_{m=0}^{n} \binom{n}{m} g_{,j}^{(m)} f_{,ik}^{(n-m)} +$$

$$\sigma \sum_{m=0}^{n} \binom{n}{m} g_{,k}^{(m)} f_{,ij}^{(n-m)} + \sigma \sum_{m=0}^{n} \binom{n}{m} g^{(m)} f_{,ijk}^{(n-m)} +$$

$$2 \sum_{m=0}^{n} \binom{n}{m} f_{,jk}^{(m)} g_{,i}^{(n-m)} + 2 \sum_{m=0}^{n} \binom{n}{m} f_{,j}^{(m)} g_{,ik}^{(n-m)} +$$

$$2 \sum_{m=0}^{n} \binom{n}{m} f_{,k}^{(m)} g_{,ij}^{(n-m)} + 2 \sum_{m=0}^{n} \binom{n}{m} f^{(m)} g_{,ijk}^{(n-m)}$$

$$\Rightarrow g_{,ijk}^{(n)} = - \left\{ \begin{array}{l} \sigma \left(\begin{array}{l} \sum_{m=0}^{n} \binom{n}{m} g_{,jk}^{(m)} f_{,i}^{(n-m)} + \sum_{m=0}^{n} \binom{n}{m} g_{,j}^{(m)} f_{,ik}^{(n-m)} + \\ \sum_{m=0}^{n} \binom{n}{m} g_{,k}^{(m)} f_{,ij}^{(n-m)} + \sum_{m=0}^{n} \binom{n}{m} g^{(m)} f_{,ijk}^{(n-m)} \end{array} \right) + \\ 2 \left(\begin{array}{l} \sum_{m=0}^{n} \binom{n}{m} f_{,jk}^{(m)} g_{,i}^{(n-m)} + \sum_{m=0}^{n} \binom{n}{m} f_{,j}^{(m)} g_{,ik}^{(n-m)} + \\ \sum_{m=0}^{n} \binom{n}{m} f_{,k}^{(m)} g_{,ij}^{(n-m)} + \sum_{m=1}^{n} \binom{n}{m} f^{(m)} g_{,ijk}^{(n-m)} \end{array} \right) \end{array} \right\} / 2 f^{(0)}$$

$$(6)$$

The nth order time derivatives for the four-order partial derivative are given by

$$0 = \sigma \sum_{m=0}^{n} \binom{n}{m} g^{(m)}_{,jkl} f^{(n-m)}_{,i} + \sigma \sum_{m=0}^{n} \binom{n}{m} g^{(m)}_{,jk} f^{(n-m)}_{,il} + \sigma \sum_{m=0}^{n} \binom{n}{m} g^{(m)}_{,jl} f^{(n-m)}_{,ik} + \sigma \sum_{m=0}^{n} \binom{n}{m} g^{(m)}_{,j} f^{(n-m)}_{,ikl} +$$

$$\sigma \sum_{m=0}^{n} \binom{n}{m} g^{(m)}_{,kl} f^{(n-m)}_{,ij} + \sigma \sum_{m=0}^{n} \binom{n}{m} g^{(m)}_{,k} f^{(n-m)}_{,ijl} + \sigma \sum_{m=0}^{n} \binom{n}{m} g^{(m)}_{,l} f^{(n-m)}_{,ijk} + \sigma \sum_{m=0}^{n} \binom{n}{m} g^{(m)} f^{(n-m)}_{,ijlk} +$$

$$2 \sum_{m=0}^{n} \binom{n}{m} f_{,jkl} g_{,i} + 2 \sum_{m=0}^{n} \binom{n}{m} f^{(m)}_{,jk} g^{(n-m)}_{,il} + 2 \sum_{m=0}^{n} \binom{n}{m} f^{(m)}_{,jl} g_{,ik} + 2 \sum_{m=0}^{n} \binom{n}{m} f^{(m)}_{,j} g^{(n-m)}_{,ikl} +$$

$$2 \sum_{m=0}^{n} \binom{n}{m} f^{(m)}_{,kl} g^{(n-m)}_{,ij} + 2 \sum_{m=0}^{n} \binom{n}{m} f^{(m)}_{,k} g^{(n-m)}_{,ijl} + 2 \sum_{m=0}^{n} \binom{n}{m} f^{(m)}_{,l} g^{(n-m)}_{,ijk} + 2 \sum_{m=1}^{n} \binom{n}{m} f^{(m)} g^{(n-m)}_{,ijkl}$$

$$\Rightarrow g^{(n)}_{,ijkl} = - \left\{ \sigma \begin{pmatrix} \sum_{m=0}^{n} \binom{n}{m} g^{(m)}_{,jkl} f^{(n-m)}_{,i} + \sum_{m=0}^{n} \binom{n}{m} g^{(m)}_{,jk} f^{(n-m)}_{,il} + \\ \sum_{m=0}^{n} \binom{n}{m} g^{(m)}_{,jl} f^{(n-m)}_{,ik} + \sum_{m=0}^{n} \binom{n}{m} g^{(m)}_{,j} f^{(n-m)}_{,ikl} + \\ \sum_{m=0}^{n} \binom{n}{m} g^{(m)}_{,kl} f^{(n-m)}_{,ij} + \sum_{m=0}^{n} \binom{n}{m} g^{(m)}_{,k} f^{(n-m)}_{,ijl} + \\ \sum_{m=0}^{n} \binom{n}{m} g^{(m)}_{,l} f^{(n-m)}_{,ijk} + \sum_{m=0}^{n} \binom{n}{m} g^{(m)} f^{(n-m)}_{,ijlk} \end{pmatrix} + 2 \begin{pmatrix} \sum_{m=0}^{n} \binom{n}{m} f^{(m)}_{,jkl} g^{(n-m)}_{,i} + \sum_{m=0}^{n} \binom{n}{m} f^{(m)}_{,jk} g^{(n-m)}_{,il} + \\ \sum_{m=0}^{n} \binom{n}{m} f^{(m)}_{,jl} g^{(n-m)}_{,ik} + \sum_{m=0}^{n} \binom{n}{m} f^{(m)}_{,j} g^{(n-m)}_{,ikl} + \\ \sum_{m=0}^{n} \binom{n}{m} f^{(m)}_{,kl} g^{(n-m)}_{,ij} + \sum_{m=0}^{n} \binom{n}{m} f^{(m)}_{,k} g^{(n-m)}_{,ijl} + \\ \sum_{m=0}^{n} \binom{n}{m} f^{(m)}_{,l} g^{(n-m)}_{,ijk} + \sum_{m=1}^{n} \binom{n}{m} f^{(m)} g^{(n-m)}_{,ijkl} \end{pmatrix} \right\} / 2 f^{(0)}$$

(7)

APPENDIX C: TRANSFORMATIONAL VARIABLES FOR J_2 TRAJECTORY AND STATE TRANSITION TENSOR CALCULATIONS.

Four variables need to be introduced to allow the binomial form of Leibnitz product rule to be used for generating arbitrary order time and partial derivative models. Leibnitz product rule is used to assemble arbitrary order time derivative models for products of variables. The resulting mathematical identities are very simple to derive and code, and the basic algorithm scales to more complicated problems. The resulting algorithms are ideally suited for massive parallel implementations.

Step 1: Simplify the z*z term

Variable Definition:

$$z2 = z * z$$

Arbitrary Order Time Derivative:

$$z2^{(n)} = \sum_{m=0}^{n} \binom{n}{m} z^{(m)} z^{(n-m)}$$

Partial Derivatives and Arbitrary Order Time Derivatives:

$$z2_{,i}^{(n)} = \sum_{m=0}^{n} \binom{n}{m} \left(z_{,i}^{(m)} z^{(n-m)} + z^{(m)} z_{,i}^{(n-m)} \right)$$

$$z2_{,ij}^{(n)} = \sum_{m=0}^{n} \binom{n}{m} \left(z_{,ij}^{(m)} z^{(n-m)} + z_{,i}^{(m)} z_{,j}^{(n-m)} + z_{,j}^{(m)} z_{,i}^{(n-m)} + z^{(m)} z_{,ij}^{(n-m)} \right) \tag{1}$$

$$z2_{,ijk}^{(n)} = \sum_{m=0}^{n} \binom{n}{m} \begin{pmatrix} z_{,ijk}^{(m)} z^{(n-m)} + z_{,ij}^{(m)} z_{,k}^{(n-m)} + z_{,ik}^{(m)} z_{,j}^{(n-m)} + z_{,i}^{(m)} z_{,jk}^{(n-m)} + \\ z_{,jk}^{(m)} z_{,i}^{(n-m)} + z_{,j}^{(m)} z_{,ik}^{(n-m)} + z_{,k}^{(m)} z_{,ij}^{(n-m)} + z^{(m)} z_{,ijk}^{(n-m)} \end{pmatrix}$$

Step 2: Simplify the z*z*x term:

Variable Definition:

$$z2x = z * z * x$$

Arbitrary Order Time Derivative:

$$z2x^{(n)} = \sum_{m=0}^{n} \binom{n}{m} z2^{(m)} x^{(n-m)}$$

Partial Derivatives and Arbitrary Order Time Derivatives:

$$z2x_{,i}^{(n)} = \sum_{m=0}^{n} \binom{n}{m} \left(z2_{,i}^{(m)} x^{(n-m)} + z2^{(m)} x_{,i}^{(n-m)} \right)$$

$$z2x_{,ij}^{(n)} = \sum_{m=0}^{n} \binom{n}{m} \left(z2_{,ij}^{(m)} x^{(n-m)} + z2_{,i}^{(m)} x_{,j}^{(n-m)} + z2_{,j}^{(m)} x_{,i}^{(n-m)} + z2^{(m)} x_{,ij}^{(n-m)} \right)$$

$$z2x_{,ijk}^{(n)} = \sum_{m=0}^{n} \binom{n}{m} \left(\begin{array}{l} z2_{,ijk}^{(m)} x^{(n-m)} + z2_{,ij}^{(m)} x_{,k}^{(n-m)} + z2_{,ik}^{(m)} x_{,j}^{(n-m)} + z2_{,i}^{(m)} x_{,jk}^{(n-m)} + \\ z2_{,jk}^{(m)} x_{,i}^{(n-m)} + z2_{,j}^{(m)} x_{,ik}^{(n-m)} + z2_{,k}^{(m)} x_{,ij}^{(n-m)} + z2^{(m)} x_{,ijk}^{(n-m)} \end{array} \right)$$

Step 3: Simplify the z*z*y term:

Variable Definition:

$$z2y = z * z * y$$

Arbitrary Order Time Derivative:

$$z2y^{(n)} = \sum_{m=0}^{n} \binom{n}{m} z2^{(m)} y^{(n-m)}$$

Partial Derivatives and Arbitrary Order Time Derivatives:

$$z2y_{,i}^{(n)} = \sum_{m=0}^{n} \binom{n}{m} \left(z2_{,i}^{(m)} y^{(n-m)} + z2^{(m)} y_{,i}^{(n-m)} \right)$$

$$z2y_{,ij}^{(n)} = \sum_{m=0}^{n} \binom{n}{m} \left(z2_{,ij}^{(m)} y^{(n-m)} + z2_{,i}^{(m)} y_{,j}^{(n-m)} + z2_{,j}^{(m)} y_{,i}^{(n-m)} + z2^{(m)} y_{,ij}^{(n-m)} \right)$$

$$z2y_{,ijk}^{(n)} = \sum_{m=0}^{n} \binom{n}{m} \left(\begin{array}{l} z2_{,ijk}^{(m)} y^{(n-m)} + z2_{,ij}^{(m)} y_{,k}^{(n-m)} + z2_{,ik}^{(m)} y_{,j}^{(n-m)} + z2_{,i}^{(m)} y_{,jk}^{(n-m)} + \\ z2_{,jk}^{(m)} y_{,i}^{(n-m)} + z2_{,j}^{(m)} y_{,ik}^{(n-m)} + z2_{,k}^{(m)} y_{,ij}^{(n-m)} + z2^{(m)} y_{,ijk}^{(n-m)} \end{array} \right)$$

Step 4: Simplify the z*z*z term:

Variable Definition:

$$z3 = z2 * z$$

Arbitrary Order Time Derivative:

$$z3^{(n)} = \sum_{m=0}^{n} \binom{n}{m} z2^{(m)} z^{(n-m)}$$

Partial Derivatives and Arbitrary Order Time Derivatives:

$$z3_{,i}^{(n)} = \sum_{m=0}^{n} \binom{n}{m} \left(z2_{,i}^{(m)} z^{(n-m)} + z2^{(m)} z_{,i}^{(n-m)} \right)$$

$$z3_{,ij}^{(n)} = \sum_{m=0}^{n} \binom{n}{m} \left(z2_{,ij}^{(m)} z^{(n-m)} + z2_{,i}^{(m)} z_{,j}^{(n-m)} + z2_{,j}^{(m)} z_{,i}^{(n-m)} + z2^{(m)} z_{,ij}^{(n-m)} \right)$$

$$z3_{,ijk}^{(n)} = \sum_{m=0}^{n} \binom{n}{m} \left(\begin{array}{l} z2_{,ijk}^{(m)} z^{(n-m)} + z2_{,ij}^{(m)} z_{,k}^{(n-m)} + z2_{,ik}^{(m)} z_{,j}^{(n-m)} + z2_{,i}^{(m)} z_{,jk}^{(n-m)} + \\ z2_{,jk}^{(m)} z_{,i}^{(n-m)} + z2_{,j}^{(m)} z_{,ik}^{(n-m)} + z2_{,k}^{(m)} z_{,ij}^{(n-m)} + z2^{(m)} z_{,ijk}^{(n-m)} \end{array} \right)$$